

Aus dem Institut für Landwirtschaftliche Verfahrenstechnik
der Christian-Albrechts-Universität zu Kiel

**(Automatische) Detektion, Posenbestimmung und
Identifizierung von Pferden mit Verfahren der 2D- und
3D-Bildverarbeitung unter Verwendung biometrischer
Merkmale**

Dissertation

zur Erlangung des Doktorgrades
der Agrar- und Ernährungswissenschaftlichen Fakultät
der Christian-Albrechts-Universität zu Kiel



vorgelegt von
Dipl.-Ing. Henry Stahl
aus Hamburg

Kiel, November 2015

Name, Vorname:	Stahl, Henry
Immatrikulations-Nr:	1008039
Promotionsfach:	Agrarwissenschaft / Verfahrenstechnik Tierproduktion
Dekan:	Prof. Dr. habil. Eberhard Hartung
1. Gutachter:	Prof. Dr. habil. Eberhard Hartung
2. Gutachter:	Prof. Dr. habil. Joachim Krieter
Institut:	Institut für Landwirtschaftliche Verfahrenstechnik
In Kooperation:	Fachbereich Technik der Fachhochschule Westküste
Betreuung Arbeitsgruppe:	Prof. Dr. Kristina Schädler
Beginn am:	08.12.2010
Tag der mündlichen Prüfung:	27.01.2016

Danksagung

Diese Dissertation entstammt meiner Tätigkeit als Wissenschaftlicher Mitarbeiter am Fachbereich Technik der Fachhochschule Westküste. Unter der Leitung von Frau Prof. Dr. Kristina Schädler war ich in dem Projekt „Automatische Erkennung von Individuen bei Nutztieren: ANIMET“ eingesetzt. Das Projekt „ANIMET“ bildete den Grundstein für das Thema meiner Promotion. Nach Projektende konnte meine Arbeit im Institut MaViTec an der Fachhochschule Westküste und durch ein anschließendes Stipendium des Instituts für Landwirtschaftliche Verfahrenstechnik der Universität Kiel weitergeführt werden.

Daher möchte ich zum einen Prof. Dr. Kristina Schädler für die Betreuung, die enge Zusammenarbeit und ihre Hilfsbereitschaft, die deutlich über das Projekt „ANIMET“ hinaus ging, herzlichst danken. Zum anderen danke ich meinem Doktorvater Prof. Dr. Eberhard Hartung für die ebenfalls weitreichende großartige Betreuung. Ohne die Unterstützung dieser beiden, im fachlichen sowie im persönlichen Bereich, wäre diese Arbeit nicht zustande gekommen. Für die gute und unkomplizierte Zusammenarbeit im Fachbereich Technik der Fachhochschule Westküste möchte ich Prof. Dr. Reiner Nawrath und Prof. Dr. Stephan Hußmann meinen Dank aussprechen. In diesem Zuge danke ich ebenfalls den Mitarbeitern Sebastian Bodó, Kevin Harksen und Prof. Dr. Torsten Edeler für die anregenden Gespräche. Außerdem bedanke ich mich bei den studentischen Hilfskräften, die mir beim Erfassen der Daten und dem Markieren tatkräftig zur Seite standen. Der Aufbau einer Versuchsstation war für die Grundlage dieser Arbeit unabdingbar. Daher gilt mein Dank der Firma HIT. Besonders zu erwähnen ist die gute Kooperation mit Thorsten, „Kolle“ und Susanne Hinrichs, die mir den Umgang mit den Pferden nahe gebracht haben und mir eine große Hilfe beim Erfassen der Daten waren. Zudem möchte ich Jan Znaniecki für den Stallumbau und die technische Assistenz im Stall danken.

Mein besonderer Dank gilt außerdem meiner Familie und meinen Freunden, die mich in dieser Zeit motiviert und unterstützt haben. An dieser Stelle danke ich meiner Mutter Mercedes Vick für das unermüdliche Markieren zahlloser Pferdeköpfe. Die großartigste Unterstützung kam jedoch von meiner Frau Rabea, die mir in allen Lebenslagen beistand, unerschütterlich an mich geglaubt hat und mich liebevoll wie selbstverständlich motivierte. Danke für diese große Rücksichtnahme und deine Geduld.

Reinfeld, im November 2015
Henry Stahl

Diese Dissertation entstand
aus einer **Kooperation**



vom Fachbereich Technik
der Fachhochschule Westküste

und



dem Institut für Landwirtschaftliche Verfahrenstechnik
der Christian-Albrechts-Universität zu Kiel.

GEFÖRDERT VOM



Bundesministerium
für Bildung
und Forschung

und



ZUKUNFTSprogramm
Schleswig-Holstein
Investition in Ihre Zukunft

Kurzfassung

Die Identifikation von Individuen beim Menschen anhand von Gesichtsaufnahmen wurde bereits in zahlreichen Arbeiten behandelt. Mit aktuellen Methoden werden sogar Erkennungsraten erreicht, die den Fähigkeiten des Menschen zur Unterscheidung einzelner Personen gleich kommen. Dagegen umfasst die Literatur nur wenige Arbeiten, welche entsprechende Ansätze der Gesichtserkennung beim Menschen zur Unterscheidung von Nutztieren untersuchen. Zudem konnten keine Arbeiten ausgemacht werden, die eine Detektion der Köpfe von Tieren (Nutztiere wie Rind, Pferd und Schaf) mit seitlich am Kopf ausgerichteten Augen behandeln. Ein entsprechendes System zur Identifikation von Nutztieren anhand ihrer Gesichter per Kamera stellt eine Alternative zur Erkennung durch am Tier befestigte RFID-Transponder zum Monitoring der Tiere in der Präzisionstierhaltung (PLF) dar. Das Hauptziel der vorliegenden Arbeit war die Entwicklung und Anwendung von Methoden, die es weitestgehend automatisch ermöglichen Individuen bei Pferden anhand von Kameradaten der Gesichter zu unterscheiden. Dazu wurde ein Bilderfassungssystem aufgebaut, um unter realen Bedingungen in einer Futterstation, neben den Grauwertdaten zweier Industriekameras, die Tiefendaten der Szene zu erfassen. In dem Pferdestall, in dem sich die Tiere frei bewegen konnten, wurde eine Datenbank der Besuche einer kleinen Gruppe von Pferden mit insgesamt 587 k Frames erstellt. Die Methoden zur Detektion und Posennormalisierung der Pferdeköpfe in den Bilddaten sind speziell für die Pferde angepasst und entwickelt worden. Für das Erzeugen von Vergleichsdaten und Kostenfunktionen zur Optimierung der Parameter wurden eigene Implementierungen erstellt. Die Detektion der Pferdeköpfe arbeitet ausschließlich auf den Tiefendaten, um unabhängig von Fellfarbe und -zeichnung der Tiere zu sein, und erreicht auf den Vergleichsdaten eine Treffergenauigkeit von 97,4 %. Die Posenbestimmung, mit der die anschließende Posennormalisierung durchgeführt wird, arbeitet ebenfalls auf den Tiefendaten und ist auf die Kopfform der Pferde angepasst. Die Posennormalisierung erzeugt eine virtuelle Ansicht des Grauwertbildes vom Kopf des Tieres, welche unabhängig von der tatsächlichen Ausrichtung des Kopfes unter Berücksichtigung der bestimmten Pose und der individuellen Kopfform erstellt wird. Die Identifikation von Individuen aus einer Gruppe von neun Pferden erreicht mit dem hier angewendeten Verfahren der Eigengesichter bereits mit geringer Auflösung und einem Lernanteil von 12,5 % der Bilddaten (25 Bilder pro Tier) eine Treffergenauigkeit von 97 %. Diese Arbeit zeigt somit einen erfolgreich unter den realen Bedingungen in einem Pferdestall umsetzbaren Weg auf, der eine weitestgehend automatische Identifikation von Pferden mit allen nötigen Vorverarbeitungsschritten der Bildverarbeitung unter Verwendung der biometrischen Merkmale im Gesicht der Tiere beschreibt.

Abstract

The recognition of human individuals based on facial images is a wide spreaded topic within research literature. Modern face recognition methods achieve recognition rates approaching human-level performance. Whereas there's only very limited literature/research on the adaptation of methods to recognize individuals in livestock. Additionally no sound scientific literature deals with the detection of those animals' heads in livestock (e.g. cattle, horse or sheep), whose eyes are placed laterally on their heads. An appropriate system being able to do animal identification on face images is a true alternative to current systems using radio-frequency identification (RFID) transponders for animal monitoring in precision livestock farming (PFL).

The main objective of the current research work was to develop and apply methods to identify individual horses only based on facial camera data, which should run at the greatest possible extent automatically. To achieve this goal a device has been developed that uses multiple sensors for capturing 2D and 3D data under real-life conditions within a feeding station for horses. The developed device allows the capturing of depth data of the scene in addition to grey scale data of two industrial cameras. A database of a small group of horses has been created during their visits to an automatic feeding station inside the housing. Altogether 587k frames have been captured. Special methods were developed and/or adapted to detect the individual heads of the horses and to normalize their pose in the image data automatically. Additionally, special implementations were compiled to generate comparative data as well as specific cost functions.

The detection of the horses' heads is based exclusively on the depth data, in order to be independent of color and pattern on the animal's coat. It achieves an accuracy (ACC) of 97,4% on the comparative data. Pose determination, as well as the preceding step towards pose normalization, perform exclusively on the depth data and were customized towards the horses' head shapes. The used pose normalization creates a virtual view of the horses' head grey scale images, which, regardless of the actual head's orientation, takes into account the determined pose and the individual shape of the head to create the normalized view.

Identification has been performed with the method of eigenfaces on a group of nine horses reaching an accuracy (ACC) of 97% using low image resolution and 12,5% of the image data (25 images per animal) for training. The current research work was successful in showing a way to create an image processing which enables the almost completely automatic detection and identification of horses by utilizing the biometric features within the animals' faces under real-life conditions.

Inhaltsverzeichnis

1. Einleitung	1
1.1. Problemstellung	2
1.2. Zielsetzung	4
2. Kenntnisstand	5
2.1. Bildverarbeitung im Kontext von Tieren	5
2.1.1. Bildverarbeitung im Bereich von Wildtieren	5
2.1.2. Bildverarbeitung im Bereich von landwirtschaftlichen Nutztieren	6
2.2. Detektion	9
2.2.1. Detektion von Objekten	9
2.2.2. Detektion von Menschen	10
2.2.3. Detektion von Wildtieren	12
2.2.4. Detektion von landwirtschaftlichen Nutztieren	16
2.3. Posenbestimmung und Normalisierung	18
2.3.1. Posenbestimmung und Normalisierung bei Menschen	18
2.3.2. Posenbestimmung und Normalisierung bei Wildtieren	23
2.4. Identifikation	25
2.4.1. Biometrische Identifikation beim Menschen	25
2.4.2. Gesichtserkennung zur Identifikation von Menschen	26
2.4.3. Bildverarbeitungsmethoden zur Identifikation von Wildtieren	33
2.4.4. Identifikationsmethoden von landwirtschaftlichen Nutztieren	35
3. Material und Methoden	39
3.1. Eingesetzte Tiere	39
3.2. Versuchsstall / Messanordnung	42
3.3. Bilderfassung	46
3.3.1. Synchrone Datenaufnahme	48
3.3.2. Kamerakalibrierung	52
3.4. Detektion	64
3.4.1. Ebenenparameter in 3D Daten zur Segmentierung	65
3.4.2. Clustern durch Hough-„Voting“	71
3.4.3. Segmentierung durch Winkel der Ebenennormalen	79
3.4.4. Detektion von Kopfkandidaten aus segmentierten Bilddaten .	90
3.5. Posenbestimmung und Normalisierung	92
3.5.1. Grobe Posenbestimmung	92
3.5.2. Posenverfeinerung und Landmarkendetektion	98

3.5.3.	Normalisierung der 2D Daten anhand der 3D Informationen .	112
3.6.	Identifikation der einzelnen Individuen bei Pferden	122
3.6.1.	Vorverarbeitung durch Normalisierung der Grauwerte	122
3.6.2.	Eigengesichter auf 2D Daten	125
3.7.	Versuchsdurchführungen	130
3.7.1.	Genauigkeitsmessung der Tiefendaten	130
3.7.2.	Aufbau Datenbank und Aufnahmen der Pferde	137
3.7.3.	Vergleichsdaten	142
3.7.4.	Orientierung der Pferdeköpfe in den Vergleichsdaten	149
3.7.5.	Parameterwahl zur Detektion und Detektionsrate	152
3.7.6.	Parameter zur Posenbestimmung und Landmarkendetektion .	160
3.7.7.	Auswahl der Daten und Vorgehen zur Identifikation	167
3.8.	Datenaufbereitung und Datenauswertung	173
3.8.1.	Datenauswertung mittels Boxplots	173
3.8.2.	Datenauswertung mittels ROC-Grafik	175
3.8.3.	CMC-Kurve	182
4.	Ergebnisse	187
4.1.	Aufnahme von 2D/3D Daten in einer Versuchsstation	187
4.2.	Ergebnisse zur Messgenauigkeit der Tiefenkamera	189
4.3.	Orientierung der Pferdeköpfe	195
4.4.	Ergebnisse der Detektion	199
4.4.1.	Parametervergleich	199
4.4.2.	Detektionsraten	204
4.4.3.	Parameterwahl	213
4.5.	Ergebnisse der Posenbestimmung und Normalisierung	219
4.5.1.	Parametervergleich	219
4.5.2.	Parameterwahl	221
4.5.3.	Parameter zur Normalisierung der 2D Daten	223
4.6.	Ergebnisse der Identifizierung individueller Pferde	224
4.6.1.	Daten zur Identifikation	224
4.6.2.	Identifikationsergebnisse	232
5.	Diskussion und Fazit	245
5.1.	Aufnahme von 2D/3D Daten in der Versuchsstation	245
5.1.1.	Aufnahmeeinheit und Messanordnung im Versuchsstall	245
5.1.2.	Kamerakalibrierung und Tiefenmessungen	247
5.2.	Orientierung der Pferdeköpfe	248
5.3.	Detektion	250
5.3.1.	Segmentierung und Regionenmarkierung	251
5.3.2.	Detektion der Pferdeköpfe	254
5.4.	Posenbestimmung, Landmarkendetektion und Normalisierung	259
5.5.	Identifikation	263

6. Zusammenfassung	269
A. Angaben nach dem Pferdepass der Tiere	271
B. Projektive Geometrie	277
C. Rekonstruktion der Raumkoordinaten aus den Disparitätenwerten	281
C.1. Umkehr der Projektion	282
D. Registrieren von 3D Daten durch den Iterative Closest Point - Algorithmus	285
D.1. Iterative Closest Point - Algorithmus	286
E. Ergebnisse der Detektion	291
E.1. Parametervergleich	291
E.1.1. Parametervergleich I	291
E.1.2. Parametervergleich II	295
E.2. Parameterwahl	305
E.2.1. Parameterwahl I	305
E.2.2. Parameterwahl II	308
F. Ergebnisse zur Parameterwahl der Posenbestimmung	311

Abbildungsverzeichnis

1.1. Bildverarbeitungskette zur automatischen Identifikation von Pferdegesichtern	2
2.1. Übersicht der Identifikationsverfahren im 2D	31
3.1. Pferde Übersicht	40
3.2. Offenstall Grundriss	42
3.3. Versuchsstation a) Aufsicht, b) Seitenansicht	44
3.4. Leitsystem a) mit V-Ausschnitt, b) bestehende Futterstation	45
3.5. Originalaufnahmen der Versuchsstation a) Rückansicht, b) Aufsicht	45
3.6. Bild der Aufnahmeeinheit	46
3.7. Hardware-Triggersignal Abnahme auf der Kinect	49
3.8. Schema der Signale am Mikrocontroller	51
3.9. Markierungen der LEDs auf den Kamerabildern	51
3.10. Perspektivische Projektion	53
3.11. Euklidische Transformation vom Welt- ins Kamerakoordinatensystem	55
3.12. Radiale Verzerrung	57
3.13. Tangentiale Verzerrung	57
3.14. Transformation zwischen Kamerakoordinatensystemen	60
3.15. Kamerabilder des Kalibrierobjektes	61
3.16. Kamerakalibrierung mit unterschiedlichen Bezugssystemen	62
3.17. Normalenvektor mit Winkeln der Einheitssphäre	68
3.18. Anpassung der Winkel des Normalenvektors in der Einheitssphäre	70
3.19. Hough-Transformation für Punkt im Koordinatenraum	72
3.20. Ebene mit entsprechendem Punkt im Parameterraum	72
3.21. Vierer-Nachbarschaft der Regionenmarkierung	74
3.22. Achter-Nachbarschaft der Regionenmarkierung	75
3.23. Übersicht der Bildverarbeitungsschritte zum Clustern der Ebenen	77
3.24. Pferdekopf Profile a) Übersicht b) transversal c) sagittal	80
3.25. Kopfformen	81
3.26. Winkelabweichung der Ebenennormalen \mathbf{n} zur Blickrichtung	82
3.27. Maskierung durch Winkelabweichungen	85
3.28. Negativbeispiel: Maskierung durch Winkelabweichungen	86
3.29. Nachbarschaft der Regionenmarkierung in Grauwertbild mit vergrößertem Pixelabstand	89
3.30. Beispiel vereinzelter Maske der Segmentierung nach Kapitel 3.4.3	93
3.31. Masseschwerpunkt des Kopfes auf 2D Maske	94

3.32. Masseschwerpunkt der Nüstern auf 2D Maske	95
3.33. Abstand der Masseschwerpunkte $\ \mathbf{M}_N - \mathbf{M}_C\ < R_d$	96
3.34. Dilatation der vereinzelter Maske	99
3.35. 3D Beispieldaten korrigiert mit grober Pose #1	99
3.36. 3D Beispieldaten korrigiert mit grober Pose #2	100
3.37. Schichten der Beispieldaten #1	101
3.38. Schichten der Beispieldaten #2	101
3.39. Kreise eingepasst in einzelne Schichten (Datensatz #1)	103
3.40. Kreiszentren mit Medianebene (Datensatz #1)	104
3.41. Bereich zur Bestimmung der Landmarken (Datensatz #1)	107
3.42. Extrahieren der Punktmengen (Datensatz #1)	108
3.43. Zentraler Bereich \mathcal{N} auf Nasenrücken (Datensatz #1)	110
3.44. Landmarken projiziert auf die Bilder der Farbkamera	112
3.45. Baryzentrische Koordinate innerhalb einer Delaunay-Triangulation .	115
3.46. Diagramm zur Bestimmung des virtuellen Bilds	117
3.47. Virtuelles Kamerabild mit Tiefenkarten [a+d) Original b+e) ohne Korrektur c+f) mit Korrektur]	119
3.48. Diagramm zur Bestimmung mehrerer virtuellen Bilder	120
3.49. Auswahl zur Bestimmung der Transferfunktion $\mathcal{T}(G_i)$	124
3.50. Virtuelles Kamerabild mit (b) und ohne (a) Normalisierung der Grau- werte	125
3.51. Rekonstruktion mit einem Eigenface	128
3.52. Rekonstruktion mit zwei Eigengesichtern	129
3.53. Skizze der Messanordnung a) Seitenansicht, b) Rückansicht, c) De- tailansicht	131
3.54. Originalaufnahmen der Messanordnung a) Rückansicht, b) Seitenansicht	132
3.55. Maske auf Infrarot-Bild an Position 350 cm (Messobjekt rot umran- det, Maske in grün)	133
3.56. Messablauf	135
3.57. Prinzip der 3D-Distanzauswertung	136
3.58. Unterschied in den Aufnahmen durch den Umbau nach 2011	140
3.59. Webinterface zum Setzen der Markierungen (Augen und Nüstern) . .	143
3.60. Maske generiert aus den Markierungen für Augen und Nüstern . . .	145
3.61. Prinzip zur Bestimmung der 3D Daten der markierten Pferdeköpfe .	146
3.62. Korrektur der Markerpositionen	148
3.63. Transformation des Modells auf die 3D Daten des aktuellen Frames .	149
3.64. Beispielfbilder ohne und mit Selbstverdeckung	151
3.65. Beispiel einer Maske durch Segmentierung (C : rot + grün) und der gewünschten Maske (M : weiß + grün)	153
3.66. Beispielgruppen (Rosi) der manuellen Gruppierung der Daten	168
3.67. Beispielgruppen (Blüte) der manuellen Gruppierung der Daten . . .	169
3.68. Schema Datenauswahl zur Kreuzvalidierung	170
3.69. Boxplot zur Auswertung der Messergebnisse	174
3.70. Wahrheitsmatrix	175

3.71. ROC-Grafik mit fünf diskreten Klassifikatoren	178
3.72. ROC-Grafik mit Kurven aus kontinuierlichen Zahlen eines Klassifikators	179
3.73. Konfusionsmatrix einer mehr-Klassen Entscheidung	183
3.74. Beispiel einer CMC-Kurve	185
4.1. Boxplots der Fehler der Tiefenkamera in unterschiedlichen Entfernungen	190
4.2. Längenfehler der horizontalen Linien (blau) des Kalibrierobjekts in unterschiedlichen Entfernungen	192
4.3. Längenfehler der vertikalen Linien (rot) des Kalibrierobjekts in unterschiedlichen Entfernungen	193
4.4. Histogramm über die Orientierungswinkel der <i>Vergleichsdaten</i>	195
4.5. Histogramm über die Orientierungswinkel der <i>Vergleichsdaten</i> mit und ohne Selbstverdeckung	196
4.6. Orientierungswinkel zur Bestimmung von Selbstverdeckung	197
4.7. ROC- und PR-Kurven der Testdatensätze 1-2	205
4.8. ROC- und PR-Kurven der Testdatensätze 3-5	206
4.9. ROC- und PR-Kurven der Testdatensätze 6-8	207
4.10. ROC- und PR-Kurven der Testdatensätze 9-11	208
4.11. ROC- und PR-Kurven der Testdatensätze 12-14	209
4.12. ROC- und PR-Kurven des Testdatensatz 15	210
4.13. Frames pro Gruppe mit Ausschlussbedingung	230
4.14. Frames pro Gruppe vor und nach Ausschlussprüfung	231
4.15. Die 9 Pferde der Test- und Validierungsgruppe	232
4.16. Identifikationsergebnisse der zweiten Industriekamera	233
4.17. Identifikationsergebnisse der zweiten Industriekamera (vergrößert) .	234
4.18. CMC-Kurve der Identifikation (zweite Industriekamera) mit 36 Eigenwerten und einer 37,5% Auflösung (30×66)	236
4.19. ROC- und PR-Kurven der Binärklassifikation zu jedem Individuum (2. Industriekamera, 5 und 25%)	237
4.20. ROC- und PR-Kurven der Binärklassifikation zu jedem Individuum (2. Industriekamera, 50 und 75%)	238
4.21. ROC- und PR-Kurven der Binärklassifikation zu jedem Individuum (2. Industriekamera, 97,5%)	239
4.22. Unterschied in normalisierten Grauwertbildern	241
5.1. Längenfehler der horizontalen (blau) bzw. vertikalen (grau) Linien des Kalibrierobjekts (#2 und #3) in unterschiedlichen Entfernungen . .	248
5.2. Kosten zur Parametervariation der Parameter α und t_b ($T_\psi = 9.1^\circ$ und $T_\theta = 5.6^\circ$ fest) des ersten Lerndatensatzes	255
5.3. Kosten zur Parametervariation der Parameter T_ψ und T_θ ($\alpha = 57.6^\circ$ und $t_b = 9.3$ fest) des ersten Lerndatensatzes	256
5.4. Kosten zur Parametervariation der Parameter b_ψ und b_θ des ersten Lerndatensatzes	258

A.1. Diagramme aus den Pässen der Tiere (1)	273
A.2. Diagramme aus den Pässen der Tiere (2)	274
A.3. Diagramme aus den Pässen der Tiere (3)	275
B.1. Projektive Ebene \mathbb{P}^2 ; Euklidische Ebene bei $z = 1$	277
C.1. Tiefeninformation durch Stereo-Vision	281

Tabellenverzeichnis

2.1. Übersicht biometrischer Identifikationsmerkmale beim Menschen . . .	26
3.1. Fellfarbe und Zeichnungen der Tiere	41
3.2. Gerätespezifikation Aufnahmeeinheit	47
3.3. Übersicht der eingesetzten Verfahren	78
3.4. Übersicht der Aufnahmen	141
3.5. Parameterliste des Cluster-Algorithmus	153
3.6. Parameterliste der Segmentierung aus Kapitel 3.4.3	154
3.7. Parameterliste der Detektion von Pferdeköpfen aus Kapitel 3.4.4 . .	158
3.8. Parameterliste der Groben Posenbestimmung aus Kapitel 3.5.1 . . .	160
3.9. Parameterliste der Posenverfeinerung aus Kapitel 3.5.2	163
3.10. Beschränkung der Parameter zur Posenverfeinerung aus Kapitel 3.5.2	164
3.11. Beispieldatensatz zur Beschreibung der ROC-Grafik	179
3.12. Punktkoordinaten in der ROC-Grafik	180
4.1. Übersicht der Anzahl erfasster Frames (Besuche) pro Tier	188
4.2. Fehlerbereiche in der Übersicht	191
4.3. Horizontale Längenfehler in der Übersicht	194
4.4. Vertikale Längenfehler in der Übersicht	194
4.5. Wahrheitsmatrix nach Selektion mittels Winkelerorientierung	198
4.6. Parameter zum Clustern aus Kapitel 3.4.2	199
4.7. Ergebnisse der Segmentierung aus Kapitel 3.4.3	200
4.8. Ergebnisse der Detektion nach dem Clustern aus Kapitel 3.4.2 . . .	203
4.9. Ergebnisse der Detektion nach der Segmentierung aus Kapitel 3.4.3 .	203
4.10. Treffergenauigkeit ACC der Detektion/Segmentierung (Kapitel 3.4.3)	211
4.11. Treffergenauigkeit ACC der Detektion/Segmentierung (Kapitel 3.4.3)	211
4.12. Treffergenauigkeit ACC der Detektion/Clustern (Kapitel 3.4.2) . . .	212
4.13. Treffergenauigkeit ACC der Detektion/Clustern (Kapitel 3.4.2) . . .	212
4.14. Parameterübersicht der Segmentierung aus Kapitel 3.4.3	214
4.15. Parameterübersicht der Detektion nach der Segmentierung	216
4.16. Ergebnisübersicht der Detektion mit einheitlichem \mathcal{H}_T	217
4.17. Parameter zur <i>Grobe Posenbestimmung</i> aus Kapitel 3.5.1	219
4.18. Parameter zur Posenverfeinerung aus Kapitel 3.5.2	220
4.19. Parameterübersicht A zur <i>Grobe Posenbestimmung</i> aus Kapitel 3.5.1	221
4.20. Parameterübersicht B zur Posenbestimmung aus Kapitel 3.5.2	221
4.21. Ergebnisübersicht der Posenbestimmung der Kapitel 3.5.1 und 3.5.2	222
4.22. Übersicht der Anzahl detektierter Frames mit Pferdekopf	225

4.23. Übersicht der Anzahl gruppierter Frames (1)	226
4.24. Übersicht der Anzahl gruppierter Frames (2)	227
4.25. Konfusionsmatrix zur Identifikation zwischen den Individuen #1 . . .	240
4.26. Konfusionsmatrix zur Identifikation zwischen den Individuen #2 . . .	242
4.27. Konfusionsmatrix zur Identifikation zwischen den Individuen #3 . . .	242
4.28. Konfusionsmatrix zur Identifikation zwischen den Individuen #4 . . .	243
4.29. Konfusionsmatrix zur Identifikation zwischen den Individuen #5 . . .	243
A.1. Geburtsdatum und Geschlecht der Tiere	271
A.2. Ursprung der Tiere mit Lebensnummern	272
A.3. Vermerk zum Kopf unter „Abzeichen bei Fuß der Mutter“	272
B.1. Transformationen im \mathbb{P}^2 und \mathbb{P}^3	279
E.1. Parameter (1) der Segmentierung aus Kapitel 3.4.3	291
E.2. Parameter (2) der Segmentierung aus Kapitel 3.4.3	292
E.3. Parameter (3) der Segmentierung aus Kapitel 3.4.3	292
E.4. Parameter (4) der Segmentierung aus Kapitel 3.4.3	293
E.5. Parameter (5) der Segmentierung aus Kapitel 3.4.3	293
E.6. Parameter (6) der Segmentierung aus Kapitel 3.4.3	294
E.7. Parameter (7) der Segmentierung aus Kapitel 3.4.3	294
E.8. Parameter (1) der Detektion nach dem Clustern aus Kapitel 3.4.2 . . .	295
E.9. Parameter (2) der Detektion nach dem Clustern aus Kapitel 3.4.2 . . .	296
E.10. Parameter (3) der Detektion nach dem Clustern aus Kapitel 3.4.2 . . .	296
E.11. Parameter (4) der Detektion nach dem Clustern aus Kapitel 3.4.2 . . .	297
E.12. Parameter (5) der Detektion nach dem Clustern aus Kapitel 3.4.2 . . .	297
E.13. Parameter (6) der Detektion nach dem Clustern aus Kapitel 3.4.2 . . .	298
E.14. Parameter (1) der Detektion nach der Segmentierung aus Kapitel 3.4.3	299
E.15. Parameter (2) der Detektion nach der Segmentierung aus Kapitel 3.4.3	299
E.16. Parameter (3) der Detektion nach der Segmentierung aus Kapitel 3.4.3	300
E.17. Parameter (4) der Detektion nach der Segmentierung aus Kapitel 3.4.3	300
E.18. Parameter (5) der Detektion nach der Segmentierung aus Kapitel 3.4.3	301
E.19. Parameter (6) der Detektion nach der Segmentierung aus Kapitel 3.4.3	301
E.20. Ergebnisse (1) der Detektion nach der Segmentierung aus Kapitel 3.4.3	302
E.21. Ergebnisse (2) der Detektion nach der Segmentierung aus Kapitel 3.4.3	302
E.22. Ergebnisse (4) der Detektion nach der Segmentierung aus Kapitel 3.4.3	303
E.23. Ergebnisse (5) der Detektion nach der Segmentierung aus Kapitel 3.4.3	303
E.24. Ergebnisse (6) der Detektion nach der Segmentierung aus Kapitel 3.4.3	304
E.25. Ergebnisse (7) der Detektion nach der Segmentierung aus Kapitel 3.4.3	304
E.26. Ergebnisübersicht (1) der Segmentierung	305
E.27. Ergebnisübersicht (2) der Segmentierung	306
E.28. Ergebnisübersicht (3) der Segmentierung	307
E.29. Ergebnisübersicht (1) der Detektion nach der Segmentierung	308
E.30. Ergebnisübersicht (2) der Detektion nach der Segmentierung	309

E.31. Ergebnisübersicht (3) der Detektion nach der Segmentierung	310
F.1. Ergebnisübersicht (1) der Posenbestimmung aus Kapitel 3.5.1 und 3.5.2312	
F.2. Ergebnisübersicht (2) der Posenbestimmung aus Kapitel 3.5.1 und 3.5.2312	
F.3. Ergebnisübersicht (3) der Posenbestimmung aus Kapitel 3.5.1 und 3.5.2313	
F.4. Ergebnisübersicht (4) der Posenbestimmung aus Kapitel 3.5.1 und 3.5.2313	
F.5. Ergebnisübersicht (5) der Posenbestimmung aus Kapitel 3.5.1 und 3.5.2314	

Abkürzungen

Kürzel	Bedeutung	Beschreibung
AAM	<i>active appearance model</i>	-
ACC	<i>accuracy</i>	Treffergenauigkeit
AdaBoost	<i>adaptive boosting</i>	adaptives Boosting (adaptives Verstärken)
AUC	<i>area under curve</i>	Fläche unter der ROC-Kurve
BCS	<i>body condition score</i>	Körperkonditionsbeurteilung
CDF	<i>cumulative distribution function</i>	Summenverteilungsfunktion
CIHONV	<i>coordinate invariant histogram of normal vectors</i>	-
CMC	<i>cumulative match curve</i>	Identifikationsergebnis-Kurve nach Rang
CNN	<i>convolutional neural network</i>	Künstliches Neuronales Netz
CoHOG	<i>co-occurrence histograms of oriented gradients</i>	-
DCS	<i>discriminant color space</i>	Diskriminierender Farbraum
DCT	<i>discrete cosine transformation</i>	Diskrete Kosinustransformation
DCV	<i>discriminant common vectors</i>	-
DLT	<i>direct-linear-transformation</i>	Direkt lineare Transformation
DNA	Deoxyribonucleic acid	-
DNN	<i>deep neural networks</i>	Tiefe Neuronale Netze
DPM	<i>deformable part model</i>	Verformbares Teilmodell
EBG	<i>elastic bunch graph</i>	Elastische Bündelgraphen
EBGM	<i>elastic bunch graph matching</i>	Verfahren zum Vergleich der EBG
EGI	<i>extended gaussian image</i>	Erweitertes Gaußsches Bild
EP	<i>evolutionary pursuit</i>	Evolutionäres Verfahren
EVG	Elektronisches Vorschaltgerät	Frequenzumwandler für Leuchtstoffröhren
EW	Eigenwert(e)	-
FDF	<i>four-directional features</i>	Vier-Richtungs Feature
FLD	<i>fisher linear discriminant</i>	Entspricht hier der LDA
FN	<i>false negative</i>	Klassifikation: Falsch-Negativ
FPGA	<i>field programmable gate array</i>	Integrierter Schaltkreis (IC)
FP	<i>engl: false positiv</i>	Klassifikation: Falsch-Positiv
FPR	<i>false positive rate</i>	Falsch-Positiv-Rate (auch Ausfallrate - <i>fallout</i> oder Falschalarm-Rate - <i>false alarm rate</i>)

Kürzel	Bedeutung	Beschreibung
FPS	Frames pro Sekunde	Anzahl der Bilder, die in einer Sekunde aufgenommen werden
GIC	<i>gamma intensity correction</i>	Gamma-Korrektur der Intensitäten im Bild
GLCM	<i>grey level co-occurrence matrix</i>	-
GPS	<i>generalized pattern search</i>	Suchalgorithmus
HF	homomorphe Filter	-
HOG	<i>histogram of oriented gradients</i>	Richtungshistogramme
HONV	<i>histogram of oriented normal vectors</i>	Histogramm zur Winkelorientierung der Normalenvektoren
HOOG	<i>haar of oriented gradients</i>	-
ICA	<i>independent component analysis</i>	Unabhängigkeitsanalyse
ICP	<i>iterative closest point</i>	Registrierungsverfahren
IR	Infrarot	Wellenlängenbereich des Lichtes ($780nm - 1000\mu m$)
ISOMAP	<i>isometric transformation</i>	Isometrische Transformation
JPEG	Joint Photographic Experts Group	Bildformat mit verlustbehafteter Kompression
KLT	Kanade-Lukas-Tomansi	Algorithmus zur Merkmalsverfolgung - <i>feature tracker</i>
KNN	<i>k-nearest-neighbour</i>	Methode der k-Nächsten-Nachbarn
KPCA	<i>kernel principal component analysis</i>	Kernel PCA
LBP	<i>local binary pattern</i>	-
LDA	<i>linear discriminant analysis</i>	Lineare Diskriminanzanalyse
LED	Licht emittierende Diode	Leuchtdiode: Halbleiter-Bauelement welches Licht emittiert
L_2 -Norm	Euklidische Norm	-
L_∞ -Norm	Maximumnorm	-
LUT	<i>lookup-table</i>	Umsetzungstabelle
MHI	<i>motion histogram image</i>	Bewegungshistogramm-Bild
NIR	Nah-Infrarot	Unterbereich des IR Bereiches (siehe IR) ($780nm - 1400nm$)
NN	<i>nearest-neighbour(hood)</i>	Verfahren der Nächsten-Nachbarn
PCA	<i>principal component analysis</i>	Hauptkomponentenanalyse
PDF	<i>probability density function</i>	Wahrscheinlichkeitsdichtefunktion
PLF	<i>precision livestock farming</i>	Präzisionstierhaltung

Kürzel	Bedeutung	Beschreibung
PNG	<i>portable network graphics</i>	Bildformat mit verlustfreier Kompression („portable Netzwerkgrafik“)
PPV	<i>positive predictive value</i>	Genauigkeit - <i>precision</i> (auch positiver Vorhersagewert)
PR-Kurve	<i>precision-recall curve</i>	-
RFID	<i>radio-frequency identification</i>	Identifikation über elektromagnetische Wellen
RFID-Tag	<i>radio-frequency identification tag</i>	Chip zur Identifikation über RFID
RGB-Bild	Rot, Grün, Blau im Bild	3 Kanälbild mit den Farben Rot, Grün und Blau
ROC	<i>receiver-operator-characteristics</i>	Grenzwertoptimierungskurve
ROR	<i>rank one recognition rate</i>	Erkennungsrate des ersten Ranges
SA	<i>simulated annealing</i>	Suchalgorithmus der simulierten Abkühlung
SIFT	<i>scale-invariant feature transformation</i>	Skalierungsinvariante Merkmals-transformation
SOM	<i>self-organizing map</i>	Selbstorganisierenden Karten
SRC	<i>sparse representation classifier</i>	Klassifikator aus spärlicher Repräsentation
SURF	<i>speeded-up robust features</i>	„Beschleunigte, robuste Merkmale“
SVM	<i>support vector machine</i>	Stützvektormaschine
TN	<i>true negative</i>	Klassifikation: Richtig-Negativ
TNR	<i>true negative rate</i>	Richtig-Negativ-Rate (auch Spezifität - <i>specificity</i>)
TOF	<i>time-of-flight</i>	Laufzeit (Kameratyp ähnlich Laser-scanner)
TP	<i>true positiv</i>	Klassifikation: Richtig-Positiv
TPR	<i>true positive rate</i>	Richtig-Positiv-Rate (auch Trefferquote - <i>recall</i> oder Sensitivität bzw. Empfindlichkeit - <i>sensitivity</i>)
UHF	<i>ultra high frequency</i>	Ultrahochfrequenz (915 MHz)
USB	<i>universal serial bus</i>	Universelles serielles Bussystem
UV	Ultraviolett	Wellenlängenbereich des Lichtes (10nm – 380nm)
UWB	<i>ultra wide band</i>	Ultraweitband
VG	Vordergrund	-

1. Einleitung

Die automatische Identifikation von landwirtschaftlichen Nutztieren anhand von Bilddaten gewinnt immer mehr an Bedeutung, ist bisher jedoch in nur wenigen wissenschaftlichen Arbeiten behandelt worden. Ganz anders verhält es sich mit dem Identifizieren von menschlichen Gesichtern. Hier erreichen Methoden der Bildverarbeitung bereits ein Niveau die Personen in Bilddaten zu identifizieren, welches den Fähigkeiten des Menschen gleich kommt [Taigman u. a. 2014]. Neben den Identifikationsalgorithmen, welche eine Klassifikation zwischen den Individuen durchführt, wird zur automatischen Identifikation von Gesichtern eine Detektion (das Auffinden der Gesichter im Bild) und eine Posennormalisierung (Ausrichten der Bilddaten oder der eingesetzten Filter) benötigt. Die jeweiligen Verfahren, welche beim Menschen zur erfolgreichen Gesichtidentifikation aus beliebigen Bilddaten (keine reine Identifikation) eingesetzt werden, lassen sich jedoch nicht ohne weiteres auf jedes Tier anwenden. So finden sich zum Teil große Unterschiede zwischen den Tierarten, welches ein Grund sein mag, weshalb nur wenige Arbeiten zur Gesichtsdetektion bei Tieren zu finden sind [Zhang u. a. 2008]. Viele dieser Arbeiten befassen sich mit Tieren, die dem Menschen ähnliche Strukturen im Gesicht aufweisen (z.B. Schimpansen [Loos u. Ernst 2013]) oder aber ausgeprägte Ohren und frontal ausgerichtete Augen haben (z.B. Katzen und Panda Zhang u. a. [2011]). Diese Bedingungen sind jedoch nicht auf die in der vorliegenden Arbeit untersuchten Pferde und viele weitere landwirtschaftliche Nutztiere (z.B. Rinder, Schweine und Schafe) zu übertragen.

Vorrangig in landwirtschaftlichen Betrieben ist ein durchgehendes Monitoring immer größerer Nutztier-Herden gefragt, denn eine unzureichende Überwachung kann Einfluss auf die Tiergesundheit (z.B. spät erkannte Krankheiten) und die Produktionsleistung in den Betrieben haben [Khoramshahi u. a. 2014]. Die Präzisionstierhaltung (PLF - *precision livestock farming*) beabsichtigt durch entsprechende Sensorik die sich verändernden Bedingungen im Betrieb zu erfassen und diese möglichst einzeltierspezifisch auszuwerten bzw. zu nutzen. Dies reicht von der Futteraufnahme und der möglichst frühzeitigen Diagnose von Krankheitserscheinungen bis hin zur Klimatisierung und Belüftung in den Ställen. Hierbei ist nach Wathes u. a. [2008] die Präzisionstierhaltung (PLF - *precision livestock farming*) im besten Fall auf das Tier-Individuum anzuwenden, was bedingt durch die Kosten derselben nicht immer möglich ist.

1.1. Problemstellung

Das Erfassen individueller Messwerte je Nutztier zum Steuern der Umgebungsfaktoren nach der Präzisionstierhaltung (PLF - *precision livestock farming*) ist auf Sensoren oder Transponder (z.B. RFID-Tag) zur eindeutigen Einzelidentifikation angewiesen. Diese werden in der aktuellen Tierhaltung meist an jedem Tier befestigt, um eine Unterscheidung zwischen den Tieren zu ermöglichen. Alternativ hierzu können lokale Sensoren, wie Kameras, im Stall genutzt werden. Die Auswertung digitaler Bilddaten kann in vielen Gebieten der Präzisionstierhaltung (PLF - *precision livestock farming*) genutzt werden [Berckmans 2006; Khoramshahi u. a. 2014]. Der Einsatz von Kamerasystemen hat gegenüber am Tier befestigten Sensorsystemen den Vorteil, dass die Tiere durchgehend beaufsichtigt werden können ohne diese zu beeinflussen oder sogar zu stören [Berckmans 2006] (z.B. Anbringen, Überprüfen oder schlechter Sitz der Sensoren, etc.). Zudem wird die Technik zum Aufzeichnen digitaler Videos immer besser und günstiger.

Offen ist jedoch die Problemstellung, innerhalb einer Gruppe von Nutztieren einzelne Individuen anhand der Bilddaten ohne weitere Marker automatisch unterscheiden zu können (Detektion und Identifikation), um die in der Tierhaltung erfassten Daten dem jeweiligen Einzeltier automatisch zuzuordnen zu können. Hierzu bietet sich die Unterscheidung der Tiere anhand ihrer Gesichter durch den erfolgreichen Einsatz beim Menschen an. Abbildung 1.1 stellt den auf das Pferd, das in dieser Arbeit untersuchte Nutztier, übertragenen Ablaufgraphen von der Aufnahme über die Detektion bis hin zur Identifikation der Gesichter dar, woraus sich gleichzeitig die nacheinander zu bearbeitenden Arbeitsschritte der vorliegende Dissertation bilden. Das mit der Abbildung 1.1 beschriebene Vorgehen hat sich bei der Identifikation von menschlichen Gesichtern bewährt [Lu u. a. 2003] und wurde in vergleichbarer Form bereits bei Schimpansen eingesetzt [Loos u. Ernst 2013].

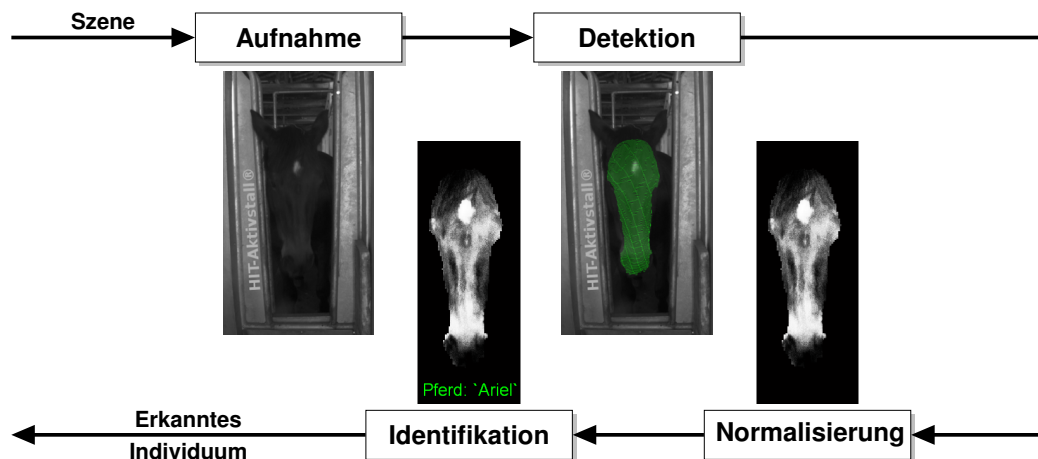


Abbildung 1.1.: Bildverarbeitungskette zur automatischen Identifikation von Pferdeg Gesichtern [Verändert nach: Loos u. Ernst 2013]

Die erfolgreiche automatische Identifikation (letzter Schritt aus Abbildung 1.1) ist demnach auf die jeweils vorangehenden Aufzeichnungs- und Bildverarbeitungsschritte angewiesen, die in der vorliegenden Arbeit auf das Pferd zu übertragen und anzupassen sind. Dadurch ergeben sich weitere grundsätzliche Problem- bzw. Aufgabenstellungen.

- Bei den Pferden handelt es sich um Fluchttiere, die sehr schreckhaft sein können und durch ihre Halslänge einen im Vergleich zum Menschen großen Bewegungsfreiraum haben ihren Kopf im Raum zu positionieren. Eine fehlende Kooperation der Tiere erschwert das Aufzeichnen von Bilddaten. Zudem gibt es keine Forschungsarbeiten die sich mit der rechnergestützten Unterscheidung von Pferdegesichtern befassen, weshalb auch auf keine bestehende Datenbank mit verwertbaren Bilddaten zurückgegriffen werden konnte.
- Die unterschiedlichen Farben und Zeichnungen im Fell der Pferde, wie sie auch bei Rindern auftreten [Anguzza 2013], erschweren es die Tiere vom Hintergrund zu trennen sowie eine Trennung von Kopf und Rumpf der Tiere auf den Bilddaten durchzuführen.
- Die genannten Fellfarben und -zeichnungen erschweren das Auffinden von Landmarken bei sich optisch unterscheidenden Tieren. Die Landmarken sollten zur Normalisierung der aufgenommenen Tiergesichter auch bei stark unterschiedlichen Pferdegesichtern an vergleichbaren Positionen sein.
- Verfahren zur Detektion und Posenbestimmung zur Normalisierung (bzw. zudem Landmarkendetektion) der Gesichter von Nutztieren auf Tiefendaten würden die auf den verschiedenen Fellfarben und -zeichnungen basierenden Problemstellungen umgehen. Entsprechende Verfahren konnten in der Literatur jedoch nicht aufgefunden werden.

1. Einleitung

1.2. Zielsetzung

Hauptziel dieser Arbeit ist die Klärung der Möglichkeiten einer weitestgehend automatischen Identifikation von Individuen bei Pferden anhand von Bilddaten der Tierköpfe innerhalb einer Futterstation unter realen Bedingungen. Aus dem in Abbildung 1.1 dargestellten Vorgehen zur Identifikation von Pferden anhand von Aufnahmen der Köpfe ergeben sich folgende Arbeitsschritte bzw. Teilziele:

- Konzeption und Aufbau eines Bilderfassungssystems, welches in einem Pferdestall unter praxistypischen Bedingungen das Erfassen der Bilddaten vom Kopf der Tiere ermöglicht.
 - Hierbei ist ein automatisches Auslösen ohne zusätzliche Sensoren, wie einer Lichtschranke, anzustreben.
 - Bei Mehrkamera-Systemen ist ein synchrones Erfassen der Daten sowie eine Kalibrierung der Kameras zueinander erforderlich.
 - Zudem sind Tier und Technik zu schützen, so dass weder die Pferde Gefahr laufen sich zu verletzen, als auch die Technik nicht von den Tieren manipuliert (zerstören, verstellen, verschieben oder verschmutzen der Kameras) werden kann.
- Automatische bzw. automatisierte Detektion der Pferdeköpfe in den Bilddaten.
 - Das Auffinden der Köpfe innerhalb der Bilder sollte ohne das vorherige Wissen um Fellzeichnung und -farbe der Individuen ermöglicht werden, um eine Detektion zuvor unbekannter Tiere zu ermöglichen.
- Automatische bzw. automatisierte Normalisierung der Bilddaten, um eine gute Ausgangslage für die Identifikation zu erreichen. Die Normalisierung soll die unterschiedlichen Posen des Kopfes ausgleichen.
- Evaluation der Identifikation mit einem Standardverfahren der menschlichen Gesichtserkennung (Identifikations-Algorithmus) auf den unter realen Bedingungen aufgezeichneten Daten der Pferdeköpfe.
 - Vergleich der Erkennungsleistung bei Variation der Auflösung in den Bilddaten, der Lernmenge sowie der Auflösung des Gesichtsraums (Dimension der genutzten Merkmale).

Die folgenden Kapitel dieser Arbeit orientieren sich jeweils an den Einzelschritten, die sich aus dem in Abbildung 1.1 dargestellten Vorgehen zur Identifikation anhand von Gesichtsaufnahmen ergeben und welche auf die Pferde übertragen wurden.

2. Kenntnisstand

2.1. Bildverarbeitung im Kontext von Tieren

2.1.1. Bildverarbeitung im Bereich von Wildtieren

Dieses Kapitel beschreibt eine Zusammenstellung von Arbeiten, in denen Wildtiere unterschieden und Tierarten erkannt oder verfolgt und Bewegungstrajektorien generiert werden. Bis auf Ausnahmen ([Burghardt u. a. 2004a] und [Wawerla u. a. 2009]) werden jedoch keine einzelnen Tiere erkannt und vom Hintergrund getrennt, worauf im Kapitel 2.2 eingegangen wird. Entweder wird eine *Detektion* (Kapitel 2.2) oder Markierung der Tiere angenommen (z.B. [Kumar u. Divya 2014]), es wird nur ein Tier im Bild angenommen (z.B. [Hannuna u. a. 2005; Hannuna 2007]) oder die Bildverarbeitung bezieht sich auf eine gesamte Population (z.B. [Xu u. a. 2009]).

Ćalić u. a. [2005] stellt mehrere Arbeiten im Zusammenhang mit Wildaufnahmen vor. Darunter ein Verfahren zur Detektion von Tierköpfen (Löwen) von Burghardt u. a. [2004a] (siehe Kapitel 2.2.3), ein Verfahren zur Detektion und Gangerkennung von Vierbeinern [Hannuna u. a. 2005; Hannuna 2007] und die Identifikation von afrikanischen Pinguinen anhand der individuellen Punktmuster auf deren Brust [Burghardt u. a. 2004b] (siehe Kapitel 2.4.3). Wawerla u. a. [2009] stellt ein System vor, welches Biologen bei der Auswertung von Aufnahmen mit Bären unterstützt. Zur Detektion der Bären in den Videoaufnahmen werden in [Wawerla u. a. 2009] Bewegungs-Shapelets angelernt, die Ähnlichkeiten mit dem Verfahren von Viola u. Jones [2001a, b]; Viola u. a. [2003] zur Objekt-Detektion (siehe Kapitel 2.2.1) haben. In [Viola u. Jones 2004] wird dieses ([Viola u. Jones 2001a, b; Viola u. a. 2003]) zur Detektion von menschlichen Gesichtern (siehe Kapitel 2.2.2) eingesetzt. In der Arbeit von Tweed u. Calway [2002] wird ein Bewegungsmodell genutzt, um die periodische Bewegung von Tieren zu detektieren und diese in Videosequenzen aufzufinden sowie zu verfolgen (*tracking*). Gibson u. a. [2003] generiert ein Bewegungsmodell von Vierbeinern aus einem spärlichen Satz von Punkten, die über eine Bildsequenz verfolgt werden (KLT-Punktverfolger; Kanade-Lukas-Tomansi *feature tracker*). In der Arbeit von Kumar u. Divya [2014] wird die Bildverarbeitung dazu genutzt unterschiedliche Tierarten in Bildmaterial zu klassifizieren. Spampinato u. a. [2010, 2014] klassifizieren unterschiedliche Fischarten anhand von Videosequenzen und extrahieren deren Bewegungstrajektorien. Xu u. a. [2009] erstellt eine Aktivitätsschätzung, die an Fischen ausgewertet wird, anhand der Bewegungsdistanz von aufeinander folgenden Frames in Abhängigkeit der Länge des jeweiligen Tieres.

Afkham u. a. [2008], Seifzadeh u. Faez [2014] und Kumar u. Divya [2014] stellen unterschiedliche Verfahren zur Unterscheidung von Tierarten in 2D Bilddaten vor.

2. Kenntnisstand

Afkham u. a. [2008] erstellt eine Art Wörterbuch aus Grauwerten. Die Methode ordnet dabei Bilder mit Pferden der Klasse „Pferd“ erstaunlich gut zu. Jedoch wird neben dem Objekt (dem Tier) auch der Hintergrund mit zur Entscheidungsfindung berücksichtigt, so dass eine Trennung des Objekts vom Hintergrund bzw. das Lokalisieren des Objekts im Bild offen bleibt. Seifzadeh u. Faez [2014] vergleicht drei Features, die die Textureigenschaften oder eine Kombination aus Textur und Gestalt beschreiben. Zur Unterscheidung der Tierarten setzt Seifzadeh u. Faez [2014] die Methode der Nächsten-Nachbarschaft (NN - *nearest-neighbour(hood)*) ein. Kumar u. Divya [2014] vergleichen unterschiedliche Klassifikationsverfahren (Nächste-Nachbarschaft - *nearest-neighbour*, neuronales Wahrscheinlichkeitsnetz - *probabilistic neural network* und symbolische Klassifikatoren - *symbolic classifier*), mit denen die Tierarten aus 2D Bildern unterscheidbar sind.

2.1.2. Bildverarbeitung im Bereich von landwirtschaftlichen Nutztieren

Berckmans [2006] gibt eine Übersicht welche Sensoren im Bereich der landwirtschaftlichen Nutztierhaltung Anwendung finden. Die Sensoren werden genutzt, um ein „online“-Monitoring der Tiere zu realisieren, deren Wohlergehen zu kontrollieren sowie Krankheiten möglichst frühzeitig zu detektieren. Dabei kommen immer häufiger auch Verfahren der Bildverarbeitung zum Einsatz, die häufig eine kostengünstige Methode zum Tiermonitoring darstellen und die Betriebsleiter entlasten [Chung u. a. 2014]. In [Chung u. a. 2014] ist zudem eine Übersicht weiterer kamerabasierter Methoden im landwirtschaftlichen Umfeld zusammengetragen. Rushen u. a. [2012] stellt eine Vielzahl von Sensoren und Einsatzmöglichkeiten zur Automatisierung in Landwirtschaftsbetrieben vor. Unter anderem werden Methoden der Bildverarbeitung oder der computergestützten Geräuscherkennung beschrieben. Neben den Arbeiten zur *Detektion* aus Kapitel 2.2 soll durch die Arbeiten in diesem Kapitel die Vielzahl der Anwendungsmöglichkeiten der Bildverarbeitung in Landwirtschaftsbetrieben verdeutlicht werden. Die vorgestellten Arbeiten befassen sich vorrangig mit der Bewegungs- und Verhaltensanalyse sowie der Zustands- und Gewichtskontrolle von Nutztieren. In den Arbeiten wird wiederum (entsprechend Kapitel 2.1.1) eine *Detektion* (Kapitel 2.2) oder Markierung der Tiere angenommen (z.B. [Ahrendt u. a. 2011]), nur ein Tier im Bild angenommen (z.B. [Tillett u. a. 1997]) oder die Bildverarbeitung bezieht sich auf eine gesamte Population (z.B. [Ott u. a. 2014]).

Schweine

Wouters u. a. [1990] setzt einfache Methoden der Bildverarbeitung (Fenster Technik - *windowing*, Schwellwert und Pixelanzahl) ein, um das wärmereregulierende Verhalten (Zusammenkauern oder Ausbreiten) von Ferkeln auszuwerten und daraufhin die Umgebungstemperatur in den Schweineställen zu kontrollieren. In der Arbeit von Wouters u. a. [1990] ist es nötig, Alter bzw. Gewicht der Tiere zu kennen, denn diese

haben Einfluss auf die Anzahl der Pixel, die einem Schwein im Bild entsprechen. Shao u. a. [1997] umgeht die Angabe von Alter oder Gewicht der Ferkel, indem ein Neuronales-Netz auf dem Fourier-Transformierten Binärbild, welches angibt, welche Pixel im Bild einem Tier entsprechen, erstellt und ausgewertet wird. In [Shao u. a. 1998] werden weitere Features mit der bisherigen Methode aus [Shao u. a. 1997] verglichen. Shao u. a. [1998] schlägt eine Kombination aus zentralen Momenten, dem Umfang und der Fläche als Feature für das bereits zuvor ([Shao u. a. 1997]) genutzte Neuronale-Netz vor. Xin [1999]; Xin u. Shao [2002]; Shao [2003]; Xin u. Shao [2005] und Shao u. Xin [2008] stellen die weiterführenden Arbeiten vor, um sich bewegende Schweine von der beschriebenen Analyse auszunehmen und ein echtzeit fähiges System aufzubauen.

Tillett u. a. [1997] und Ahrendt u. a. [2011] stellen unterschiedliche Methoden zur Verfolgung (*tracking*) von Schweinen in Videosequenzen vor. Die Tiere werden in aufeinander folgenden Farbbildern, die eine Draufsicht auf eine Gruppe von Schweinen zeigen, durch eine Modellierung verfolgt. Ahrendt u. a. [2011] nutzt zudem eine Vorder- / Hintergrund Schätzung, die anhand der Farbinformation durchgeführt wird, um die Methode des Verfolgens zu verbessern. Eine vorherige Detektion (vgl. Kapitel 2.2.4) wird in den Arbeiten ([Tillett u. a. 1997; Ahrendt u. a. 2011]) nicht beschrieben.

Chung u. a. [2014] und Ott u. a. [2014] stellen Verfahren zur videobasierten Überwachung von Schweinen vor, welches automatisch die Aktivität der Tiere innerhalb einer Gruppe aufzeichnet und so deren Tagesrhythmus bestimmt als auch überwacht. Ismayilova [2014] und Viazzi u. a. [2014] nutzen zur Detektion einiger Vorzeichen aggressiven Verhaltens bei Schweinen ein Bewegungshistogramm-Bild (MHI - *motion histogram image*). Das MHI (Bewegungshistogramm-Bild) wird aus zuvor manuell gelabeltem Bildmaterial erstellt. Zudem stellt Ismayilova [2014] eine weitere Methode zur Aktivitätsmessung (vgl. [Chung u. a. 2014; Ott u. a. 2014]) aus einer Bildsequenz der Draufsicht auf eine Gruppe von Schweinen dar.

Hühner

Leroy u. a. [2006] stellt ein System zur Verhaltensbestimmung von einzeln gehaltenem Geflügel aus Kameradaten vor. Dawkins u. a. [2009, 2012] setzt einfache Webcams ein, um die Gangart durch den optischen Fluss (*optical flow*) in den Bilddaten auszuwerten. Die Auswertung findet für die gesamte Herde und nicht auf Basis einzelner Tiere statt. Aydin u. a. [2010] erstellt anhand von Videoaufnahmen eine Bewegungsauswertung von Zuchthähnchen, die mit dem Aktivitätsindex der Tiere korreliert. Die Aktivitätsindizes werden ebenfalls für eine Gruppe von Hühnern erstellt. Lee u. a. [2011] erweitert die Arbeiten [Dawkins u. a. 2009, 2012] um das Erfassen von Unruhen unter den Tieren, welches zur Vorhersage von Federpicken (gestörtes Verhalten) unter den Tieren genutzt werden kann (siehe [Rushen u. a. 2012]).

2. Kenntnisstand

Rinder

Die Arbeiten von Song u. a. [2008]; Leroy u. a. [2008]; Bahr u. a. [2008]; Poikalainen u. a. [2009]; Poursaberi u. a. [2010]; Pluk u. a. [2012] und van Hertem u. a. [2014] befassen sich mit der automatischen, videobasierten Gangerkennung von Rindern. In der Arbeit von Cangar u. a. [2008] wird ein Verfahren zur automatischen Erkennung des Verhaltens von kalbenden Kühen aus Videosequenzen beschrieben. Tsai u. Huang [2014] erkennt zudem Brunft- und Paarungsverhalten von Kühen in Bildsequenzen.

Einsatz von Tiefeninformationen (3D)

Für den Einsatz in Landwirtschaftsbetrieben werden auch 3D Informationen verarbeitet. So wird in [Tillett u. a. 2004] ein passives Stereovision-System (Aufbau mit zwei Kameras) zur Rekonstruktion des dreidimensionalen Körpers von Schweinen verwendet. Die Rekonstruktion soll dazu dienen, Formveränderungen durch Wachstum, Gewichtsveränderung, Diäten und Haltung der Schweine zu detektieren. In [Nakarmi 2013; Nakarmi u. a. 2014] wird eine TOF-Kamera (*time-of-flight* Kamera), die die Entfernung zur Kamera anhand der Laufzeit von moduliertem Licht ermittelt, eingesetzt, um Legehennen zu verfolgen (zu „tracken“). Zusätzlich setzt Nakarmi [2013; Nakarmi u. a. 2014] eine Reihe von RFID-Antennen (*radio-frequency identification antennas*) am Boden des Geheges ein, um die Identifikation der einzelnen Tiere mittels RFID-Transpondern (*radio-frequency identification transponder*) zu ermöglichen. Anglart [2010] stellt ein automatisches Verfahren vor, welches die 3D Aufnahmen einer TOF-Kamera dazu nutzt, eine Körperkonditionsbeurteilung (BCS - *body condition score*) und das Gewicht von Schwedisch Rotbunten Rindern zu schätzen. Dabei wird in [Anglart 2010] die Problematik beim Einsatz von TOF-Kameras in Bezug auf unterschiedlich farbige Zeichnungen bei Rindern erwähnt, da die TOF-Kamera im beschriebenen Fall in dunklen Regionen unzureichend genaue Messwerte erzeugt.

Kongsro [2014] setzt die Kinect, deren Daten auch in dieser Dissertation zum Einsatz kommen (siehe Kapitel 3.3), zur Gewichtsschätzung von Schweinen ein. Die Farbe der Schweinerasse hatte dabei keinen Einfluss auf die Tiefendaten und somit auf die durchgeführte Gewichtsschätzung, so dass die Methode auch bei dunklen Tieren eingesetzt werden konnte.

2.2. Detektion

2.2.1. Detektion von Objekten

Die Detektion von Objekten ist ein äußerst weitreichendes Gebiet der Bildverarbeitung. In diesem Kapitel sind daher nur solche Arbeiten aufgeführt, die im Zusammenhang mit den in dieser Arbeit verwendeten Methoden stehen bzw. in abgewandelter Form bereits zur Detektion von Tiergesichtern eingesetzt wurden (siehe Kapitel 2.2.3). Eine Übersicht aktueller Methoden zur Objektdetektion (z.B. SIFT *scale invariant feature transformation*) findet sich in [Prasad 2012]. Juan u. Gwun [2009] vergleicht die Feature SIFT *scale invariant feature transformation*, PCA-SIFT und SURF (*speeded-up robust features* [Bay u. a. 2008]), welche ebenfalls zum Auffinden von Objekten eingesetzt werden, miteinander. Eine weitere Übersicht zu Detektions- und Klassifikations-Algorithmen von Objekten ist in der Arbeit von Zhang u. a. [2013] zu finden. In der im übrigen auch die Detektion von Pferden (jedoch ganzer Körper) aus Bildern in der Arbeit [Ferrari u. a. 2008] (bzw. [Ferrari u. a. 2010]) beschrieben wird. Ferrari u. a. [2008, 2010] stellen hierzu eine Datenbank von Pferdebildern (ganzer Körper) vor.

Viola u. Jones [2001a, b]; Viola u. a. [2003] stellen eine robuste und weit verbreitete Methode zur Detektion von Objekten auf 2D Grauwertbildern vor, die auch häufig zur Gesichtsdetektion bei Menschen eingesetzt wird [Khan u. a. 2013]. Die Methode erzeugt zunächst ein Integralbild und lernt weiche Klassifikatoren (einzeln ohne hohe Aussagekraft) an, die in mehreren Kaskaden zu einem robusten Klassifikator zusammengefasst werden. Dies wird als adaptives Boosting (adaptives Verstärken - *adaptive boosting* - AdaBoost) bezeichnet. Die eingesetzten Haar-ähnlichen (*haar-like*) Features (ähnlich den Haar-Wavelets) sind einfache rechteckige Filter, bei denen die einzelnen Regionen entweder positiv oder negativ gewertet werden und in ihrer Summe den zur Entscheidung des Klassifikators genutzten Wert darstellen. Aus einem möglichst großen Satz dieser einfachen Filter wird durch das Anlernen automatisch der zur Klassifikation nutzbare Satz extrahiert. Die Klassifikation kann durch die Kaskadenstruktur schnell angewendet werden. In [Viola u. a. 2003] wird neben den Informationen eines Bildes die Bewegung über mehrere Bilder einer Videosequenz ausgewertet, um Fußgänger zu detektieren. Weitere Arbeiten, die diese Methode zur Gesichtsdetektion einsetzen, sind in den Kapiteln 2.2.2 und 2.2.3 zu finden.

Chen u. a. [2010] nutzt die Tiefeninformation, um zunächst einzelne Objektpunkte, die sich in unterschiedlichen Schichten der Tiefe befinden, aufzufinden. Ausgehend von den Objektpunkten wird ein Region-Growing-Algorithmus angewendet, um das komplette Objekt zu markieren. Hetzel u. a. [2001] stellt eine Methode zur Erkennung von Objekten durch Histogrammvergleiche unterschiedlicher Merkmale einer Tiefenkarte vor. Es wird eine Erkennung von realen Objekten mit simulierter Verdeckungen durchgeführt, wobei freigestellte Objekte ohne Hintergrund erkannt werden. Ansätze aus [Hetzel u. a. 2001] finden in dieser Dissertation Verwendung (siehe Kapitel 3.4.1).

2. Kenntnisstand

Zudem wird in Kapitel 3.4.4 ein Ähnlichkeitsfaktor ρ beschrieben, welcher vergleichbar ist mit der in [Schiele u. Crowley 2000] beschriebenen χ^2 -Abweichung (χ^2 -divergence), die durch Hetzel u. a. [2001] zum „Histogramm Matching“ verwendet wird. Der in Kapitel 3.4.4 verwendete Offset im Histogramm (zirkuläres Verschieben/Versatz anhand des durchschnittlichen Histogrammwerts) ist jedoch, wie in dem Kapitel 3.4.4 selbst beschrieben, der Arbeit von Tang u. a. [2013] zuzuordnen. Das durch

Tang u. a. [2013] beschriebene Versetzen des Histogramms umgeht die in [Hetzel u. a. 2001] beschriebene Problematik beim Vergleich versetzter Histogramme (entsteht z.B. durch veränderten Betrachtungswinkel). Um die beschriebene Problematik zu umgehen, führt Hetzel u. a. [2001] eine wahrscheinlichkeitstheoretische Erkennung (*probabilistic recognition*) ein.

Tang u. a. [2013] stellt ein Verfahren vor, welches HONV (*histogram of oriented normal vectors*) zur Detektion und Klassifikation von Objekten in Tiefenkarten nutzt. Das Verfahren wird in Kapitel 3.4.4 näher beschrieben, wobei Tang u. a. [2013] die Features in einer Pyramidenstruktur aus mehreren Auflösungsstufen aus den Bilddaten extrahiert.

2.2.2. Detektion von Menschen

Dalal u. Triggs [2005] stellen eine Methode zur Detektion von Menschen in Einzelbildern dar, die mit den eingesetzten HOG (*histogram of oriented gradients* - Richtungshistogramme) Features vergleichbar gute Ergebnisse zu den getesteten SIFT (*scale invariant feature transformation*) Features ergeben. Das in Kapitel 3.4.4 eingesetzte Verfahren der HONV (*histogram of oriented normal vectors*) stellt eine Erweiterung des HOG (*histogram of oriented gradients*) Features vom 2D- in den 3D-Raum dar [Tang u. a. 2013].

Xia u. a. [2011] verwendet die Kinect, deren Daten auch in dieser Dissertation zum Einsatz kommen (siehe Kapitel 3.3), um menschliche Köpfe durch das Einbringen von Vorwissen (3D Modell des Kopfes und der Größenverhältnisse) in den Daten aufzufinden. Die Gemeinsamkeiten der Arbeit von Xia u. a. [2011] mit dieser Arbeit beschränken sich auf die Verwendung der 3D Daten der Kinect sowie eines Geometriemodells. Jedoch wurde der im Kapitel 3.4.3 beschriebene Ansatz durch Xia u. a. [2011] angeregt.

Detektion von menschlichen Gesichtern

In den Arbeiten [Hjelmås u. Low 2001; Yang u. a. 2002; Zhang u. Zhang 2010] ist eine Übersicht unterschiedlicher Verfahren zur Detektion von menschlichen Gesichtern zusammengestellt. Diese reichen vom einfachen Merkmal, wie der Detektion durch Gesichtsfarbe oder Textur, über aktive Formmodelle (*active shape models*) bis hin zu bildbasierten Methoden, die ohne Vorwissen direkt auf den Bilddaten angelernt werden. Yang u. a. [2002] teilt die Verfahren zur Gesichtsdetektion in vier

Kategorien: Die wissensbasierten Methoden, die Verfahren mit invarianten Merkmalen, die Methoden mit Schablonenabgleich (*template matching*) und die erscheinungsbasierten (*appearance-based*) Verfahren. Die wissensbasierten Methoden verwenden vordefinierte Regeln zur Detektion der Gesichter, die auf vorherigem Wissen basieren und vorrangig zur Lokalisation der Gesichter eingesetzt werden. Die Verfahren mit invarianten Merkmalen zielen darauf ab, Strukturen im Gesicht zu finden, die robust hinsichtlich Posen- oder Lichtveränderungen sind. Der Schablonenabgleich nutzt zuvor gespeicherte Gesicht-Templates, um in einem Bild Gesichter aufzufinden. Die erscheinungsbasierten Verfahren lernen anhand repräsentativer Trainingsdaten und sind im Allgemeinen bis dato den vorherigen Verfahren hinsichtlich der Detektionsperformance in 2D Bildern überlegen [Zhang u. Zhang 2010].

In der Übersicht von Zhang u. Zhang [2010] findet sich eine detaillierte Beschreibung der menschlichen Gesichtsdetektion und derer Variationen, die, wie in [Viola u. Jones 2004] beschrieben, die Haar-ähnlichen Features aus [Viola u. Jones 2001a, b; Viola u. a. 2003] mit dem AdaBoost-Algorithmus (weiche Klassifikatoren werden in einer Kaskadenstruktur zu einem Klassifikator angeordnet, vgl. Kapitel 2.2.1) kombiniert werden. Das Boosting (Verstärken), also das Anlernen vieler weicher Klassifikatoren, die zu einem „guten“ Klassifikator zusammengefasst werden, wird in der Arbeit von Viola u. Jones [2004] verwendet und hat sich zu dem Standardverfahren der Gesichtsdetektion in der realen (praxisnahen) Anwendung entwickelt [Zhang u. Zhang 2010]. Jedoch setzt das Boosting eine große Anzahl an Trainingsdaten voraus [Zhang u. Zhang 2010]. Variationen des Verfahrens von Viola u. Jones [2004] werden bereits zur Detektion von Gesichtern bei Tieren (z.B. Löwen [Burghardt u. a. 2004a; Burghardt u. Calic 2006a, b]) angewendet (siehe Kapitel 2.2.3). Eine Diskussion zur Anwendung des Verfahrens beim Pferd findet sich in Kapitel 5.3.

Wong u. a. [2001], als auch Li u. a. [2006], stellen Methoden zur Gesichtsdetektion vor, bei denen die Augenkandidaten durch ihren Kontrast im Grauwertbild aufgefunden werden. Die unterschiedlichen Zeichnungen der Pferde, die Positionierung der Augen an den Seiten des Kopfes [Timney u. Keil 1999] sowie kaum Kontrast (hauptsächlich schwarze Augen, kein weißer Bereich sichtbar) innerhalb der Augen machen diese oder ähnliche Methoden für Pferde unbrauchbar.

Niese u. a. [2007] findet Gesichter über deren homogene Gesichtsfarbe. Diese wird in der Methode genutzt, um mit Hilfe der 3D Information Cluster zu finden. Beim Pferd wird dies nicht ohne Weiteres umsetzbar sein, da die Tiere im Allgemeinen keine homogene Gesichtsfarbe haben. Mian u. a. [2006] stellt eine Detektion von menschlichen Gesichtern vor, die in 3D Daten ein Gesicht anhand der Nase auf findet. Bedingt durch die Anatomie, lässt sich dieses Verfahren ebenfalls nicht auf das Pferd anwenden. Die durch Niese u. a. [2007] und Mian u. a. [2006] verwendete Normalisierung ist in Kapitel 2.3 aufgeführt.

2.2.3. Detektion von Wildtieren

Dieses Kapitel fasst Arbeiten zusammen, die hauptsächlich die Detektion von Wildtieren (bzw. Tieren im Allgemeinen) behandeln. Dabei werden gesamte Körper der Tiere detektiert. Das Unterkapitel *Kopf- und Gesichtsdetektion* hingegen fasst ausführlich solche Arbeiten zusammen, die sich mit der Detektion von Köpfen oder Gesichtern von Tieren beschäftigen. Im Unterkapitel *Kopf- und Gesichtsdetektion* finden sich zudem Arbeiten zur Detektion von Haustieren (z.B. Katzen [Zhang u. a. 2008]).

Die Arbeit von Burghardt [2008] zeigt umfangreich die Detektion und Identifikation von Pinguinen und Zebras in ihrem natürlichen Umfeld. Zudem wird ausführlich die Bedeutung unterschiedlichster Fellzeichnungen und -muster in ihrer Entwicklung sowie zur Detektion und Identifikation behandelt. Sharma [2013]; Koik u. Ibrahim [2012] stellen eine vergleichbar kurze Übersicht aktueller Detektionsmethoden von Tieren aus Bilddaten zusammen.

Parihk u. a. [2013] und Sharma u. a. [2014] nutzen einen verhältnismässig einfachen Ansatz des Schablonenabgleichs (*template matching*), um auf 2D Bilddaten einer Bildsequenz Tiere zu detektieren, die durch Hintergrundsubtraktion (hier die Differenz aufeinanderfolgender Frames) als sich bewegende Objekte in den Bildern aufgefunden werden. Ramanan u. Forsyth [2003]; Ramanan u. a. [2006] stellen eine Methode vor, die autonom aus einer Videosequenz ein Modell eines sich bewegenden Objekts erzeugt und dieses verfolgt (*tracking*). Das Modell besteht aus einzelnen Patches aus denen sich das Tier in jedem Bild zusammensetzt. Zusätzlich werden die erstellten Patches anhand von Farb- und Texturinformationen mit angelerten Tieren („Zebra“, „Tiger“ und „Giraffe“) verglichen, so dass ein Detektor für die jeweilige Tierart entsteht. Khorrami u. a. [2012] lernt zunächst ein Hintergrundmodell und extrahiert anschließend Tiere aus Videosequenzen anhand der größten Bewegung mittels einem Verfahren zur Bestimmung des optischen Flusses (*optical flow*). Die Besonderheit ist, dass das Verfahren trotz eines großen Versatzes der Tiere von Frame zu Frame aufgrund niedriger Frameraten anwendbar ist. Chamoso u. a. [2014] und van Gemert u. a. [2014] stellen Systeme vor, um Tiere von einem unbemannten Fluggerät aus aufzunehmen, zu detektieren und anschließend zu zählen. Die Detektion in [Chamoso u. a. 2014] erfolgt über ein künstliches Neuronales Netz (*convolutional neural network* - CNN). Van Gemert u. a. [2014] vergleicht zur Detektion verformbare Teilmodelle (*deformable part(-based) model* - DPM) mit kombinierten Support-Vektor-Maschinen (Stützvektormaschine - *support vector machine* - SVM), um die Tiere in den Bilddaten möglichst schnell aufzufinden. Zeppelzauer [2013] und Sugumar u. Jayaparvathy [2014] stellen Systeme zur Detektion von Elefanten aus Farbbildern vor. Zeppelzauer [2013] detektiert die Elefanten in den Bildern zunächst mit einem Farbmodell und verfeinert die Detektion durch das Verfolgen der Tiere über mehrere Videoframes (*tracking*). Sugumar u. Jayaparvathy [2014] nutzen die Haar Wavelet-Transformation, um aus den Farbbildern Features zu extrahieren, die

mit Hilfe des K-Means-Algorithmus (*k-means clustering*) in einzelne Gruppen aufgeteilt werden. Die Detektion der Elefanten in den Bilddaten wird durch einen Abgleich mit einer zuvor erstellten Datenbank ermöglicht, wozu in [Sugumar u. Jayaparvathy 2014] eine optimierte Distanz-Metrik vorgestellt wird. Berg u. Forsyth [2006] verwenden Text, Farbe, Form und Textur um Tierbilder von Webseiten zu clustern, welches das Erstellen einer Bilddatenbank deutlich erleichtern kann. Kaiyan u. a. [2012] nutzen eine Kombination eines Fuzzy-C Means Clustering-Algorithmus, morphologischer Operationen und einer Blob-Analyse zur Segmentierung von Tieren in Bilddaten.

Lévesque u. Bergevin [2010] stellt ein Framework vor, welches Objekte in Einzelbildern zunächst vom modellierten Hintergrund trennt und diese in den Bildsequenzen (Videosequenz) verfolgt (*tracking*). Anschließend wird ein 3D Modell des Objekts durch das Zusammenbringen der Informationen zweier Kameras (Stereo-Vision) erzeugt, um eine Skelett-Signatur des Modells mit einer Datenbank abzugleichen und darüber die Objekte auf angelernte Tierarten zu überprüfen.

Kopf- und Gesichtsdetektion

In diesem Unterkapitel wurde darauf geachtet möglichst vollständig solche Arbeiten zusammenzustellen, in denen eine Detektion von Gesichtern bei Tieren beschrieben wird. Dies stellt den aktuellen Stand der Literatur zur *Detektion* aus Kapitel 3.4 dar, da dem Autor keine vergleichbaren Arbeiten zur *Detektion* von Gesichtern bei landwirtschaftlichen Nutztieren aus Seiten- oder Frontaufnahmen (vgl. Kapitel 2.2.4) bekannt sind. Die Arbeiten [Burghardt u. a. 2004a; Burghardt u. Calic 2006a, b] verwenden die Methode von Viola u. Jones [2004] zur menschlichen Gesichtsdetektion, welche bereits in Kapitel 2.2.2 vorgestellt wurde, mit einem erweiterten Featuresatz, um Löwengesichter in Frontalaufnahmen (bzw. seitlichen Ansichten [Burghardt u. Calic 2006a, b]) zu detektieren. Die detektierten Löwengesichter werden anschließend auf den Videosequenzen verfolgt (*tracking*), so dass semantische Rückschlüsse zum Verhalten der Tiere gemacht werden können.

Zhang u. a. [2008] stellt eine Methode zur Detektion von Katzengesichtern vor. Der Detektor wird durch eine Fusion zwei einzelner Detektoren aufgebaut, wobei ein Detektor vorrangig die Form der Gesichter (Normalisierung anhand der Ohren) berücksichtigt und ein weiterer die Textur (Normalisierung anhand der Augen). Der fusionierte Detektor zeigt dabei im Vergleich zu dem einzelnen Form- bzw. Textur-detektor und einem Detektor, welcher eine optimale Transformation zur Normalisierung der Augen und Ohren nutzt, die besten Detektionsresultate. Zhang u. a. [2008] führt zur Detektion die *Haar of Oriented Gradients* (HOOG) als Feature ein, welche ähnlich dem Ansatz von Viola u. Jones [2004] mit dem AdaBoost-Algorithmus angelernt werden.

Kozakaya u. a. [2009] zeigt eine Variation, in der Haar-ähnliche (*haar-like*) Features mit AdaBoost (siehe Kapitel 2.2.1) verwendet werden, um mögliche Katzengesichter

2. Kenntnisstand

aufzufinden. Die Detektion in [Kozakaya u. a. 2009] wird mit einer Erweiterung der *histograms of oriented gradients* (HOG - Richtungshistogramme; entsprechend [Dalal u. Triggs 2005]), den sogenannten *co-occurrence histograms of oriented gradients* (CoHOG), welche die Gradientenstärke der Richtungsgradienten (*oriented gradients*) unberücksichtigt lassen, verfeinert. Kouda u. a. [2011] setzt HOG (*histograms of oriented gradients*) Feature ein, um die Gesichter von Waschbären und Marderhunden in Bildern aufzufinden. Anschließend werden die beiden Tierarten durch Kouda u. a. [2011] anhand ihrer Gesichter diskriminiert. In [Kouda u. a. 2011] wird die komplette Bildverarbeitungskette vorgestellt, die zur Detektion der Gesichter die HOG (*histograms of oriented gradients*) Feature einsetzt, um anschließend mit einer diskreten Kosinustransformation (DCT - *discrete cosine transformation*) oder der PCA (*principal component analysis*) die einzelnen Tierarten voneinander zu unterscheiden. Es findet durch Kouda u. a. [2011] keine Identifikation des Individuums (vgl. Kapitel 2.4.3) statt.

Fleuret u. Geman [2008] nutzt stationäre Features zur Objektmodellierung und -detektion, wobei als Beispiel Katzen zur Objektdetektion gewählt wurden. Zur Objektmodellierung wird in einer *coarse-to-fine* (grob zu fein) Suche zunächst der Katzenkopf detektiert, um ausgehend von diesem in einem Modell mögliche Positionen des Körpers der Katze aufzufinden. Die Kombination der Kopf- und Körperdetektion ermöglicht durch vordefinierte Ausschlusskriterien anhand der Relation zwischen den Einzelergebnissen verbesserte Detektionsresultate. Bo [2010] verwendet Grauwertmatrizen (GLCM - *grey level co-occurrence matrix*) sowie Histogramme, um Form und Textur von Katzengesichtern zu beschreiben und diese in Bilddaten zu detektieren.

In den Arbeiten [Parkhi u. a. 2011] und [Parkhi u. a. 2012] wird zunächst ein verformbares Teilmodell (*deformable part model* - DPM, vgl. [van Gemert u. a. 2014] aus Kapitel 2.2.3) mit HOG (*histogram of oriented gradients*) Features zur Detektion von Katzen- und Hundegesichtern genutzt. In [Parkhi u. a. 2011] werden zusätzlich LBP (*local binary pattern* [Ojala u. a. 1994, 1996]) im verformbaren Teilmodell verwendet. Das aufgefundene Katzen- bzw. Hundegesicht dient Parkhi u. a. [2011, 2012] anschließend als charakteristisches Körperteil, dazu Fellmerkmale (in [Parkhi u. a. 2011] Farbe und in [Parkhi u. a. 2012] zusätzlich Textur) des jeweiligen Tieres im Bild zu extrahieren, um mit dieser Information den verbleibenden Körper des Tieres vom Hintergrund im Bild zu trennen. Parkhi u. a. [2012] ermöglicht, neben der Detektion der Tiere (Hund und Katze), die Unterscheidung zwischen den Tierarten „Hund“ und „Katze“. Außerdem wird die Bestimmung einiger angelernter Rassen (12 Katzenrassen und 25 Hunderassen) beschrieben. Chanvichitkul u. a. [2007] beschreibt eine Methode zur Unterscheidung von Hunderassen auf Basis von Gesichtsaufnahmen. In [Chanvichitkul u. a. 2007] wird zunächst ein Schablonenabgleich (*template matching*) der Ohren und der Gesichtsform zur groben Einordnung der Tiere eingesetzt. Im Anschluss daran wird die Hauptkomponentenanalyse (PCA - *principal component analysis*; siehe Eigengesichter in Kapitel 3.6.2) zur Klassifikation der Rasse angewendet.

Yamada u. a. [2011] nutzt die durch Ishii u. a. [2004] beschriebenen richtungsbasierenden Features (FDF - *four-directional features*, Vier-Richtungs Feature), um frontale Katzen- und Hundegesichter auf den Bildern einer Digitalkamera zu detektieren.

Sumner u. Ross [2012] zeigen Variationen des durch Zhang u. a. [2008] vorgestellten Verfahrens, welche Features einsetzen, die eine verbesserte Detektion erlauben sollen. In [Zhang u. a. 2011] wird der zuvor in [Zhang u. a. 2008] vorgestellte Detektor um einen „verformbaren Detektor“ erweitert und auf weitere Tiere, wie Fuchs und Gepard angewendet. Zudem wird der Verbund-Detektor (Form- und Texturdetektor) zur Unterscheidung der Tierarten Katze, Tiger, Löwe, Leopard, Panda (Bambusbär), kleiner Panda (Katzenbär), Waschbär, Wolf, Fuchs und Gepard eingesetzt. Rakesh u. a. [2012] schlägt vor, die Methode von Zhang u. a. [2011] mit der von Dalal u. Triggs [2005] zu kombinieren, um zwischen Mensch und Tier unterscheiden zu können. Rangdal u. Hanchate [2014] verwenden nur noch die durch Dalal u. Triggs [2005] eingesetzten HOG (*histogram of oriented gradients* - Richtungshistogramme) Features zur Detektion der in [Zhang u. a. 2011] verwendeten Katzengesichter, zeigen jedoch keine Detektionsergebnisse. Chen u. a. [2012a] setzt zur Detektion von Pandagesichtern (Bambusbärgesichtern) ein Bereichswachstumsverfahren (*region-growing*) in Kombination mit der topologischen Prüfung der aufgefunden Gesichtsbestandteile (Augen, Ohren und Nase) ein. Dabei wird die kontrastreiche Zeichnung der Tiere ausgenutzt.

Chen u. a. [2012b] nutzt die Methode der k -fachen Nächsten-Nachbarschaft (*k-nearest-neighbour*) in zwei Stufen mit unterschiedlicher Auflösung, um anhand der Grauwertvariation im Bild den Bereich des Kopfes mit den oberen Extremitäten von Pandabären (Bambusbären) aufzufinden. Der Kopf der Pandabären wird anschließend in dem genannten Bereich anhand des Farbhistogramms in einem vorgegebenen Satz an typischen Kopfkandidaten (*example-based head detection*) gesucht.

In der Arbeit von Ernst u. Kublbeck [2011] werden mehrere Stufen mit je einem Klassifikator in einer Art Kaskadenstruktur zur Detektion von Schimpansen- und Gorillagesichtern angelernt. In jeder Stufe wird die Detektion der Schimpansen- bzw. Gorillagesichter verfeinert, zudem werden solche Bildausschnitte, die kein Gesicht eines entsprechenden Primaten enthalten, möglichst frühzeitig erkannt, um diese nicht in den darauffolgenden Stufen zu verarbeiten. In den einzelnen Stufen werden Features aus dem Gradienten der Bilder, dem Census und der Struktur im Bild ausgewertet. Die rechenintensive Auswertung der Struktur wird erst bei erfolgreichem Passieren der vorherigen Stufen berechnet. Die Kombination eines Detektors für Schimpansengesichter mit einem für Gorillagesichter ermöglicht das Unterscheiden der beiden Spezies anhand ihrer Gesichter. In den Arbeiten [Loos u. Ernst 2012] und [Loos u. Ernst 2013] wird der in [Ernst u. Kublbeck 2011] vorgestellte Gesichtsdetektor in einem Framework zur Detektion und Identifikation von Primatengesichtern (Schimpansen) eingesetzt. Die in [Loos u. Ernst 2013] verwendeten Normalisierungs- und Identifikationsverfahren werden in den Kapiteln 2.3.2 bzw. 2.4.3 beschrieben.

2. Kenntnisstand

Schlussfolgernd kann festgestellt werden, dass die in diesem Kapitel aufgeführten Arbeiten der Literatur zur Detektion von Gesichtern bei Tieren sich auf die Anwendung bei Katzen oder katzenähnlichen Tieren (z.B. Löwe, Leopard, Panda, Fuchs, Waschbär etc.), Primaten (z.B. Gorilla und Schimpansen) und Haustieren (Hund bzw. Katze) beschränken. Die Gesichter dieser Tiere unterscheiden sich stark von denen der Pferde, da beispielsweise die Augen beim Pferd seitlich ausgerichtet sind [Timney u. Keil 1999], wodurch die Pferde als Fluchttiere einen größeren Sichtbereich zu den beschriebenen Tieren (Katzen, katzenähnliche Tiere und Primaten) besitzen. Sind die Augen jedoch frontal ausgerichtet, wie bei den Katzen, ergibt sich eine in der Frontansicht wiederholende Struktur im Gesicht zwischen unterschiedlichen Individuen der entsprechenden Tierart. Hierzu lagen keine belastbaren Angaben in der Literatur vor.

2.2.4. Detektion von landwirtschaftlichen Nutztieren

Lind u. a. [2005] beschreibt eine Methode zur Bestimmung des Fortbewegungsverhaltens von Schweinen. Die Detektion beschränkt sich auf ein einzelnes Tier im Bild, welches durch eine Hintergrundsubtraktion und eine Schwellwertbildung aufgefunden wird. Die Positionen (ähnlich dem Masseschwerpunkt des aufgefundenen Objekts „Schwein“) aufeinander folgender Bilder beschreiben die jeweilige Bewegung des Schweins in einer aufgenommenen Videosequenz.

Jover u. a. [2009] nutzt eine künstliche Farbmarkierung von Ferkeln, um diese anhand der aufgezeichneten drei Kanäle einer Farbkamera im Bild auffinden und durch die Unterscheidung der Farben innerhalb des Farbraums identifizieren zu können. Kim u. Jung [2013] hingegen nutzt aktive Markierungen, die im Fall eines Temperaturanstiegs bei Kühen durch ein aufleuchtendes Lichtsignal ein krankes Tier melden sollen. Das aktive Signal wird anschließend durch Kameras lokalisiert, um die Position des Tieres zu bestimmen und melden zu können.

Porto u. a. [2011, 2012, 2013] behandeln die Detektion von Rindern in vorgegebenen Bereichen (z.B. Futtergitter) im Stall. Durch die Anwesenheit in gewissen Bereichen des Stalls kann auf das Verhalten der Tiere geschlossen werden (z.B. Futterbereich oder Ruhezone). Die Detektion der Rinder aus einer Draufsicht wird in [Porto u. a. 2011, 2012, 2013] durch einen erweiterten Satz von Haar-ähnlichen (*haar-like*) Features (ähnlich den Haar-Wavelets) mit dem Algorithmus von Viola u. Jones [2004] durchgeführt. Die Ergebnisse betrachten lediglich die Daten zwischen 7:00 und 16:00 Uhr, da in den restlichen Aufnahmen, der Arbeit entsprechend, das Tageslicht fehlte. In der Arbeit von Anguzza [2013] wird ebenfalls der Algorithmus von Viola u. Jones [2004] angewendet, um Rinder in einem Panoramabild, welches aus einzelnen Draufsichten einer Vielzahl von Kameras erstellt wurde, aufzufinden. Alternativ wird eine Detektion über die extrahierte Kontur der Tiere beschrieben. Die durch Anguzza [2013] beschriebene Methode zur Identifikation der einzelnen Individuen bei den Kühen wird in Kapitel 2.4.4 aufgegriffen.

Khoramshahi u. a. [2014] lernt ein Neuronales-Netz an, um eine Trennung von Vorder- / Hintergrund auf den Bilddaten durchzuführen, die Schweine in einer Draufsicht zeigen. Der Vordergrund in den Bilddaten entspricht den Schweinen, die mit Farbkameras unter sichtbarem und auch infrarotem, künstlichen Licht aufgenommen wurden. Khoramshahi u. a. [2014] stellt einige weitere Arbeiten vor, die ebenfalls eine Vorder- / Hintergrund-Segmentierung einsetzen, um Schweine im Bild aufzufinden: Hu u. Xin [2000] setzen dazu eine Kombination aus Bewegungsdetektion (interframe Differenzen), Likelihood-Ratio-Test und Schattenmodellierung ein. Chen u. a. [2003, 2005] implementieren eine Durchschnitts- und Schwellwertbildungs-Routine zur Detektion von Tieren mit einem FPGA (*field programmable gate array*).

Martinez-Ortiz u. a. [2013] stellt ein videobasiertes Verfahren vor, Milchkühe zu detektieren (Köpfe der Kühe von oben) und zu verfolgen (*tracking*). Die Köpfe werden dazu mit dem Algorithmus von Viola u. Jones [2004], der die bereits in Kapitel 2.2.1 beschriebenen Haar-ähnlichen (*haar-like*) Features (ähnlich den Haar-Wavelets) einsetzt, in den Bilddaten aufgefunden. Die detektierten Kopfpositionen werden anschließend in aufeinander folgenden Frames verfolgt und mit einem Kalman Filter geglättet, so dass eine Trajektorie der Bewegung der Tiere aufgezeichnet wird. Anhand der Trajektorie des Gangs wird eine Beweglichkeitsauswertung zum Auffinden von lahmen Tieren gemacht. In Martinez-Ortiz u. a. [2013] finden sich zudem Ansätze zur Identifikation der Individuen aus Bildern (vgl. Kapitel 2.4.4) des gesamten Körpers mittels einer Hauptkomponentenanalyse (PCA - *principal component analysis*), die auch in Kapitel 3.6.2 dieser Dissertation eingesetzt wird. Eine in [Martinez-Ortiz u. a. 2013] vorgestellte Alternative zur Identifikation wird mit SIFT (*scale-invariant feature transformation*) Features durchgeführt. Die Arbeit von Martinez-Ortiz u. a. [2013] findet sich nicht im Kapitel 2.4.4 wieder, da keine weitere Auswertung der Identifikationsverfahren bei den Kühen beschrieben ist.

Die bisher in diesem Kapitel (2.2.4) aufgeführten Arbeiten (mit Ausnahme von [Kim u. Jung 2013]) verwenden eine Draufsicht auf die Tiere (Schweine oder Rinder). van Hertem u. a. [2013] vergleicht mehrere Methoden zur Vorder- / Hintergrundtrennung aus Seitenansichten von Kühen. Dies stellt, wie in der Arbeit [van Hertem u. a. 2013] beschrieben, aufgrund eines nicht gleichmässigen Hintergrund und den unterschiedlich farbigen Fellzeichnungen der Tiere eine schwierige Aufgabe dar. Als möglicher Lösungsansatz bzw. zur weiteren Untersuchung wird durch van Hertem u. a. [2013] der Einsatz von 3D Kameras empfohlen (in dieser Arbeit eingesetzt und bereits in [Stahl u. a. 2012] vorgestellt).

2.3. Posenbestimmung und Normalisierung

2.3.1. Posenbestimmung und Normalisierung bei Menschen

Das Normalisieren von Gesichtern (beim Menschen) dient nach Li u. a. [2006] dazu, unterschiedliche Bildaufnahmen des Gesichts einer Person vor der eigentlichen Gesichtserkennung (Identifikation, siehe Kapitel 2.4) in einen vergleichbaren Zustand zu bringen. Hierzu zählt den Einfluss unbrauchbarer und redundanter Informationen, wie dem Hintergrund, den Haaren, der Kleidung oder Vergleichbarem, zu reduzieren (nach [Li u. a. 2006]). Das Ergebnis der Normalisierung ist die einheitliche Darstellung des Gesichts einer Person in mehreren Aufnahmen (nach [Li u. a. 2006]). Dies betrifft nach Li u. a. [2006] die Kontur, die Größe, die Rotation und die Helligkeitsverteilung des Gesichts in den Aufnahmen. Muruganantham u. Jebarajan [2012] beschreibt, dass die Posenvariation von menschlichen Gesichtern eine hohe Variation in den Bilddaten erzeugt. Nach Muruganantham u. Jebarajan [2012] fällt die Bildvariation durch Posenveränderung von Bildern einer Person häufig sogar stärker aus, als die Bildvariation zwischen den Bildern mehrerer Personen.

Die Normalisierung der Grauwerte wird in dieser Arbeit im Kapitel 2.4.2 getrennt von der Normalisierung der Pose betrachtet, da die Normalisierung der Grauwerte ein Vorverarbeitungsschritt zur Identifikation ist, der nicht für alle Identifikationsmethoden notwendig ist (siehe Kapitel 2.4).

Yi u. a. [2013] unterteilt bestehende Methoden zur posenunabhängigen Gesichtserkennung in zwei Hauptkategorien: Die 2D- und die 3D-Methoden bzw. Hybriden (2D und 3D). Die 3D-Methoden schließen dabei 2D-Methoden mit 3D Modellen ein. Nach Yi u. a. [2013] haben die 3D-Methoden im Allgemeinen eine bessere Genauigkeit im Vergleich zu den 2D-Methoden, da die Posenvariation hauptsächlich aus einer starren Bewegung des Gesichts im dreidimensionalen Raum entsteht. Den 2D-Methoden fehlt ein Freiheitsgrad, wodurch diese Methoden (2D) häufig 2D Transformationen (z.B. stückweise affine Transformationen oder weitere Approximationen) einsetzen, um die eigentliche dreidimensionale Transformation des Gesichts zu approximieren und die Fehler durch statistische Lernverfahren zu kompensieren (siehe Yi u. a. [2013]). Diese Arbeit folgt der Unterteilung der beiden Hauptkategorien in 2D- und 3D-Methoden.

2D-Methoden

Murphy-Chutorian u. Trivedi [2009] stellen eine Übersicht bestehender Methoden (hauptsächlich 2D-Methoden) zur Normalisierung von menschlichen Köpfen zusammen. Murphy-Chutorian u. Trivedi [2009] unterteilen die Methoden dabei in solche Methoden mit Erscheinungs-Vorlagen/Schablonen (*appearance template methods*), Detektor-Array Methoden, nichtlineare Regressionsmethoden, Methoden der eingebetteten Mannigfaltigkeit (*manifold embedding methods*), Methoden mit flexiblen Modellen, geometrische Methoden, Methoden zur Verfolgung der Köpfe

(*head tracking*) und hybride Methoden. Die hohe Variation unterschiedlicher Zeichnungen im Gesicht von Pferden erschwert, ohne das Vorwissen des konkreten Aussehens der Tiere (z.B. durch vorheriges Anlernen), den Einsatz solcher Methoden, die die Erscheinung auf 2D Bilddaten zur Posenbestimmung nutzen. So lassen sich Methoden zur Posenbestimmung von Pferdeköpfen, ohne das vorherige Wissen um deren Zeichnungen, auf die geometrischen Methoden und die Methoden zur Verfolgung der Köpfe (*head tracking*) unter den 2D-Methoden aus [Murphy-Chutorian u. Trivedi 2009] beschränken. Die Geometrischen Methoden nutzen, nach der Definition aus [Murphy-Chutorian u. Trivedi 2009], lokale Features wie die Augen, den Mund und die Nase im menschlichen Gesicht, um aus deren relativen Anordnung die Pose des Gesichts zu bestimmen.

Das in dieser Arbeit benutzte Verfahren zur Posenbestimmung aus Kapitel 3.5 lässt sich, nach der beschriebenen Definition, den geometrischen Verfahren zuordnen, auch wenn in dieser Arbeit 3D Informationen zur Posenbestimmung (Kapitel 3.5) ausgewertet werden. Die folgenden Arbeiten in diesem Unterkapitel beschreiben, aufgrund der großen Vielfalt an Arbeiten zur menschlichen Gesichtsnormalisierung, daher lediglich zwei geometrische Methoden. Eine weitere Übersicht bestehender Arbeiten findet sich in [Zhang u. Gao 2009], welche neben den 2D-Methoden auch Methoden mit 3D Modellen beschreibt (siehe nächstes Unterkapitel *3D-Methoden*).

Das Verfahren von Li u. a. [2006] zur Normalisierung von menschlichen Gesichtern spiegelt sehr gut die Einzelschritte zur Landmarkendetektion (Augen vgl. Kapitel 3.5.2 und Mund), der geometrischen Normalisierung im zweidimensionalen Raum (vgl. Kapitel 3.5.1 bzw. 3.5.3) und der Normalisierung der Grauwerte (vgl. Kapitel 3.6.1) wieder. Li u. a. [2006] setzt zur Detektion der Gesichter zunächst die Methode von Viola u. Jones [2001b] (entsprechend [Viola u. Jones 2004]) mit einem erweiterten Satz von Haar-ähnlichen (*haar-like*) Features ein. Wie bereits in Kapitel 2.2.2 beschrieben, werden die Grauwerte zur Detektion der Augen ausgewertet. Die Augen werden in vorgegebenen Bereichen des zuvor aufgefundenen Gesichts angenommen. Diese Bereiche werden durch zwei Fenster (ein Fenster je Auge) repräsentiert. Die Grauwerte in diesen Fenstern werden analysiert und mit einem einfachen Schwellwertverfahren in eine Maskierung überführt. Eine horizontale und eine vertikale Projektion der Binärwerte innerhalb der maskierten Fenster lässt eine Auswahl über die Extremwerte (hier das Minimum) zu, die in Li u. a. [2006] der aufgefundenen Augenposition entspricht. Die Pose des Gesichts wird über den Mittelpunkt der beiden Augenzentren und die Orientierung der Verbindungslinie zwischen den beiden Augenzentren bestimmt. Die Normalisierung korrigiert daher zunächst eine Verschiebung entsprechend dem Mittelpunkt der Augen und rotiert das Gesicht anschließend entsprechend dem Winkel der Verbindungslinie zwischen beiden Augen. Die Skalierung der Gesichter erfolgt auf eine ähnliche Weise. Hierzu wird die Position des Mundes, ähnlich wie zuvor die Augen, in einem vorgegebenen Fenster aufgefunden. Der Abstand des Mittelpunkts der Augen zum Mund wird zur Normierung der Höhe der Gesichter und der Abstand zwischen den Augen zur Normierung der Weite der Gesichter verwendet. Zuletzt werden durch Li u. a. [2006] die

2. Kenntnisstand

Grauwerte entsprechend deren Mittelwert und Varianz im Bild auf eine einheitliche Darstellung normiert, welches den Effekt unterschiedlicher Beleuchtungssituationen bei der Gesichtserkennung reduziert (vgl. Kapitel 2.4.2).

In der Arbeit von Wong u. a. [2001] wird die Detektion mit der Normalisierung kombiniert, indem zunächst Augenkandidaten (Kandidaten, da nicht unbedingt Augen) als Landmarken aufgefunden werden. Die Augenkandidaten werden anschließend kombiniert (jeweils zwei Augen), so dass auf Gesichtskandidaten geschlossen werden kann (Abhängigkeit von Augenabstand zu Gesichtsgröße im Bild). Die Gesichtskandidaten werden normalisiert (Rotation, Translation und Grauwerte), um anschließend über eine Fitness-Funktion (Kostenfunktion; vgl. Ähnlichkeitsfaktor ρ aus Kapitel 3.4.4), welche durch eine Hauptkomponentenanalyse (PCA - *principal component analysis*; siehe Eigengesichter aus Kapitel 3.6.2) bestimmt wird, solche Gesichtskandidaten zu verwerfen, die keinem Gesicht entsprechen. Zudem wird die Symmetrie der Gesichter ausgenutzt, um eine zusätzliche Validierung der Gesichtskandidaten auf ein Gesicht durchzuführen. Zuletzt werden auf dem Gesicht weitere Gesichtsmerkmale (Augenbrauen, Nasenlöcher und Mund) ausfindig gemacht. Sollte einer dieser Merkmale nicht an den vorgegebenen Positionen (Bereiche durch Fenster auf dem Gesicht) gefunden werden, so wird der Gesichtskandidat ebenfalls verworfen. Auch in dieser Arbeit wird die Symmetrie der Pferdegesichter ausgenutzt. Anstelle der 2D Grauwertinformation (die Zeichnungen der Pferde sind nicht unbedingt symmetrisch), wie in [Wong u. a. 2001], werden die Pferdegesichter in Kapitel 3.7.7 auf die Symmetrie im dreidimensionalen Raum (Tiefeninformation vorausgesetzt) geprüft.

3D-Methoden

Yi u. a. [2013] unterteilt, neben den beschriebenen Hauptkategorien (2D- und 3D-Methoden), die 3D-Methoden der Normalisierung in die vier Unterkategorien: Die Posen-Normalisierung, die Posensynthese, die Erkennung durch Einpassen (*fitting*) und die Filtertransformationen. Die in [Yi u. a. 2013] beschriebene Methode zählt zu den Filtertransformationen, bei der die Filter an die Pose und damit an das verformte Bild des Gesichts angepasst werden. Die entsprechend der Pose transformierten Filter dienen der Merkmalsextraktion, die durch die vorherige Transformation posenunabhängig wird. Die Posen-Normalisierung, zu der die in Kapitel 3.5 beschriebene Methode dieser Arbeit zählt, erzeugt aus dem zu untersuchenden Bild eine korrigierte Frontansicht, so dass diese mit den angelernten Bildern abgeglichen werden kann. Die Posensynthese korrigiert hingegen alle angelernten Bilder der Datenbank in die Pose des aktuell zu untersuchenden Bildes. Die Erkennung durch Einpassen (*fitting*) nutzt ein 3D Modell, welches anhand von Textur und Form in die angelernten sowie das zu untersuchende Bild angepasst wird. Ein virtuelles Kamerabild, wie es in den Methoden der Posen-Normalisierung bzw. der Posensynthese erzeugt wird, wird bei der Erkennung durch Einpassen nicht erzeugt.

Chai u. a. [2004, 2005] beispielsweise verwenden ein synthetisch erzeugtes 3D Modell des menschlichen Gesichts, um an bereits aufgefundenen Landmarken im 2D Bild eine Normalisierung der Ansichten zu einer Frontalansicht zu generieren. Es wird gezeigt, dass die Korrektur der Pose eine deutliche Verbesserung bei der Identifikation der Individuen bewirkt.

In [Zhang u. Gao 2009] ist eine Übersicht von Arbeiten, die 3D-Methoden mit 3D Modellen verwenden, zusammengetragen. Die einzelnen Arbeiten sind in einer tabellarischen Übersicht mit Vor- und Nachteilen zusammengestellt. Zhang u. Gao [2009] beschreibt 3D-Methoden, die entweder generische 3D Modelle verwenden oder individuelle 3D Modelle des menschlichen Gesichts durch eine bild- bzw. featurebasierte 3D Rekonstruktion erstellen. Eine 3D Rekonstruktion wird in dieser Arbeit umgangen, indem das in Kapitel 3.3 beschriebene Bilderfassungssystem genutzt wird, 3D Informationen der Szene (der Pferde bzw. Pferdeköpfe) aufzuzeichnen. Durch Gökberk u. a. [2009] wird eine Übersicht von 3D Methoden zusammengestellt, die die 2D-Methoden (auch solche mit 3D Modellen) häufig (besonders bei extremen Lichtverhältnissen) in ihrer Genauigkeit übertreffen.

Gökberk u. a. [2009] behandelt die automatische Landmarkendetektion, die automatische Registrierung und die Merkmalsextraktion in 3D in einzelnen Kapiteln. Die Merkmalsextraktion wird in dieser Arbeit im Kapitel 2.4 behandelt. Bei der Registrierung wird in [Gökberk u. a. 2009] zwischen einer rigiden und einer nicht-rigiden Transformation unterschieden. Die Transformation starrer Körper (hier Gesichter) wird als rigide und die Transformation verformbarer Körper als nicht-rigide bezeichnet. Zur Registrierung starrer Körper wird in Gökberk u. a. [2009] der ICP-Algorithmus (*iterative closest point - algorithm*, siehe Anhang D) als Standardverfahren bezeichnet. Der ICP-Algorithmus bedarf dabei einer Initialisierung, wobei der Algorithmus bei einer schlechten Initialisierung zur inkorrekten Registrierung führen kann. Die nicht-rigide Registrierung berücksichtigt neben der affinen Transformation (siehe Tabelle B.1 im Anhang B), welche bei der Registrierung starrer Körper bestimmt wird, die Verformung von Objekten (hier Gesichter). In [Gökberk u. a. 2009] wird beschrieben, dass entsprechende Algorithmen, die die Verformung berücksichtigen, häufig ein gemeinsames Modell (annotiertes Gesichtsmodell) einsetzen, auf welches die individuellen Gesichter angepasst und registriert werden. Bei menschlichen Gesichtern können diese Verfahren eingesetzt werden, um, neben der Pose, spezifische Gesichtsausdrücke des menschlichen Gesichts in einem verformten Modell nachzustellen (z.B. [Blanz u. Vetter 2003]).

Eine nicht-rigide Registrierung wurde für die in dieser Arbeit untersuchten Pferde nicht verwendet, da der Mensch mit seinen 44 unterschiedlichen Muskeln deutlich mehr Muskeln im Gesicht und dem Haarboden hat als Tiere (wie die Pferde) [DeMello 2012]. Der Mensch ist mit seinen Gesichtsmuskeln in der Lage deutlich mehr Emotionen durch Gesichtsausdrücke darzustellen als entsprechende Tiere (ausgenommen Primaten) [DeMello 2012]. Die Möglichkeiten von Pferden zu Gesichtsausdrücken bei Schmerz sind durch Glerup u. a. [2015] untersucht worden. Glerup u. a. [2015] beschreibt vorrangig Augen und Nüstern, neben den Ohren, als die sich

2. Kenntnisstand

bewegenden Teile im Gesicht der Pferde, die bei der Erkennung von Schmerzen anhand der Pferdegesichter dienen.

Im Rahmen dieses Kapitels werden weiter einige wenige Arbeiten zur Normalisierung behandelt, die als geometrische Methoden mit Posen-Normalisierung einzuordnen sind bzw. als Bezug auf die verwendete Methode erwähnenswert sind (beispielsweise die Landmarkendetektion bzw. Symmetrieebenen).

Die automatische Landmarkendetektion ist auch im Dreidimensionalen keine leichte Aufgabe (siehe [Gökberk u. a. 2009]). Die Nase ist nach Gökberk u. a. [2009] ein gutes Gesichtsmerkmal beim Menschen, welches sich durch heuristische Methoden schnell auffinden lässt, da sie sich anhand der Tiefeninformation gut vom menschlichen Gesicht abhebt. Die Nase wird in vielen Arbeiten (z.B. [Malassiotis u. Strintzis 2004; Chang u. a. 2005]) als markantes Merkmal zur Registrierung / Normalisierung von menschlichen Gesichtern eingesetzt. Wird neben der Nase, als Gesichtsmerkmal, die Symmetrieebene des Gesichts gefunden, so sind nach Gökberk u. a. [2009] die Augen und der Mund durch ihre relative Position im Gesicht recht einfach aufzufinden. Das Auffinden der Symmetrieebene kann jedoch ohne entsprechende Einschränkung durch vorher aufgefundene Landmarken sehr rechenaufwendig werden.

Trotzdem kann das Auffinden der Symmetrieebene im Gesicht zur Posenbestimmung genutzt werden. Die Symmetrieebene beschreibt die Pose jedoch nicht komplett, da zwei Translationen und eine Rotation verbleiben (siehe [Combes u. a. 2008]). Pan u. a. [2006] findet die Symmetrieebene des Gesichts mit Hilfe des erweiterten Gaußschen Bildes (EGI - *extended gaussian image*). Zur Bestimmung der Symmetrieebene wird die Gaußsche Oberflächenkrümmung auf den 3D Daten berechnet. Dies wird auf den Tiefendaten der Kinect aufgrund einer schlechteren Tiefenauflösung (siehe [Li u. a. 2013]) zu schlechteren Ergebnissen führen. Zudem haben entsprechende Methoden nach Combes u. a. [2008] Einschränkungen bei teilverdeckten Gesichtern. Combes u. a. [2008] stellt eine erweiterte Methode zum Auffinden der Symmetrieebene von bilateralen Objekten in deren 3D Punktwolken dar, bei der eine abgewandelte Form des ICP-Algorithmus (siehe Kapitel D) eingesetzt wird. In dieser Arbeit wird in Kapitel 3.5.2 eine als Medianebene Π_S bezeichnete Ebene anhand eines individuell parametrisierten, geometrischen Modells des jeweiligen Pferdekopfes bestimmt. Die beschriebene Medianebene Π_S ist jedoch auf Grundlage des Modells und nicht einer direkten Symmetrie bestimmt. Die Symmetrie der Pferdegesichter wird in dieser Arbeit erst nach der eigentlichen Posenbestimmung (Kapitel 3.5) in Kapitel 3.7.7 zur Auswahl der Pferdeköpfe geprüft, so dass die Symmetrieebene bereits durch die Pose vorgegeben ist. Die Arbeiten von Pan u. a. [2006] und Combes u. a. [2008] sind daher als mögliche Alternativen zu der in dieser Arbeit beschriebenen Methode (Kapitel 3.5) anzusehen.

McCool u. a. [2007] stellt eine Methode zur Normalisierung von menschlichen Gesichtern vor, in der die Landmarken (Augen) vorgegeben sind. Zur Registrierung setzt McCool u. a. [2007] (wie Malassiotis u. Strintzis [2004], s.o.) den ICP-Algorithmus ein, der im Anhang D beschrieben und in dieser Arbeit zum Anlernen (siehe

Kapitel 3.7.4) der vorgestellten Algorithmen verwendet wird. Niese u. a. [2007] verwendet ebenfalls den ICP-Algorithmus zur Posenbestimmung, wobei ein auf eine Person zugeschnittenes 3D Modell eingesetzt wird. In [Mian u. a. 2006] wird der ICP-Algorithmus nur zur Validierung der Posenkorrektur verwendet. Die Pose wird in [Mian u. a. 2006] über eine Hotelling Transformation bestimmt. Anschließend wird, wie auch in Kapitel 3.5.3, eine frontale Ansicht des Gesichts für die spätere Identifikation virtuell erzeugt.

Li u. a. [2013] beschreibt den Einsatz der Kinect, deren Tiefendaten auch in dieser Arbeit verwendet werden (siehe Kapitel 3.3), zur Gesichtserkennung von Menschen. Die Detektion der Nasenspitze (nach Gökberk u. a. [2009] ein gutes Gesichtsmerkmal) wird vorausgesetzt, um das Gesicht auszuschneiden. Anschließend wird der bereits erwähnte ICP-Algorithmus (siehe Kapitel D) zur Posenbestimmung eingesetzt. Die Symmetrie des Gesichts wird daraufhin dazu genutzt Fehlstellen im Gesicht auszubessern. In [Li u. a. 2013] wird ein Resampling (Neu-Abtastung) der 3D Daten dazu genutzt, die Daten zu glätten und einen einheitlichen Abtastabstand zu erhalten. Alle weiteren Schritte der Arbeit von Li u. a. [2013] zur Gesichtserkennung werden im Kapitel 2.4 erläutert.

2.3.2. Posenbestimmung und Normalisierung bei Wildtieren

Der Großteil der Arbeiten aus Kapitel 2.4.3 und dem Unterkapitel *Gesichter* des Kapitels 2.4.4, die die Identifikation von Tieren anhand von Bilddaten beschreiben, nutzen manuell gesetzte Landmarken zur Normalisierung der Bilddaten (z.B. van Tienhoven u. a. [2007]; Hiby u. a. [2009]; Kim u. a. [2005b]; Corkery u. a. [2007]; Cai u. Li [2013]) bzw. als Vorauswahl (z.B. Rechteck Lahiri u. a. [2011]).

In den Arbeiten [Loos u. Ernst 2012, 2013] und [Sandwell u. a. 2013] wird der bereits zur Detektion (siehe Kapitel 2.2.3) der Schimpansengesichter verwendete Algorithmus von Ernst u. Kublbeck [2011] mit AdaBoost (siehe Kapitel 2.2.1) zum Auffinden der Augen innerhalb der Gesichtsregion eingesetzt. Burghardt u. Campbell [2007] nutzt ein 3D Modell, um aus einzelnen Patches, die im Bild aufgefunden werden und deren Position im 3D Modell bekannt sind, ein zweidimensionales Texturabbild (*texture map*) der Brust von afrikanischen Pinguinen zu erstellen. Auf das durch eine Rückprojektion generierte Texturabbild (*texture map*) wird ein adaptives Schwellwertverfahren angewendet, um das markante Punktemuster auf dem hellen Brustbereich der Pinguine zu extrahieren und als Landmarken für die Identifikation auszuwerten.

Die meisten Arbeiten zur Identifikation von Tieren nutzen beim Normalisieren der zweidimensionalen Bilddaten eine Affine Transformation im zweidimensionalen Raum (siehe Anhang B), die sich aus Skalierung, Rotation, Translation und einer Scherung zusammensetzt. Häufig wird die Scherung ignoriert, da sich die damit ergebene Ähnlichkeitstransformation (4 Freiheitsgrade, vgl. Anhang B) mittels zweier Punkte auf

2. Kenntnisstand

dem Bild bestimmen lässt. Damit wird angenommen, dass das Objekt (das Tier) linear, steif und zweidimensional ist (siehe van Tienhoven u. a. [2007]). Dabarera u. Rodrigo [2010] bzw. Corkery u. a. [2007] nutzen die Affine Transformation anhand der zuvor markierten Augen, um die Orientierung der Elefanten- bzw. Rindergesichter vor der Identifikation (siehe Kapitel 2.4.3) in den Bildern zu normalisieren. In der Arbeit von Loos [2012] wird ein Ansatz von Huang u. a. [2007] mit dem Namen „congealing“ (zusammenlaufend) zur Normalisierung der Gesichter von Schimpansen und Gorillas eingesetzt. „Congealing“ bestimmt automatisch eine Affine Transformation für jedes zuvor markierte oder detektierte Gesicht zur Normalisierung, wobei eine Reihe von Bildern bzw. Teilausschnitte (z.B. über Rechtecke markiert) einer Klasse in eine einheitliche Position gebracht wird. Cai u. Li [2013] nutzen zur Normalisierung der Gesichter von Rindern die Methode von Peng u. a. [2010], die ebenfalls automatisch eine Affine Transformation für die Bilder erzeugt, auf denen zwei Bildpunkte als zu normalisierende Bildpunkte markiert sind. Der Algorithmus von Peng u. a. [2010] ist theoretisch in der Lage eine perspektivische Projektion (entspricht der Projektiven Transformation im 2D, vgl. Anhang B) für jedes Bild zu bestimmen (nicht in [Cai u. Li 2013] eingesetzt).

Loos u. Ernst [2012, 2013] normalisieren Schimpansengesichter durch eine perspektivische Projektion (Projektive Transformation im 2D, vgl. Anhang B), die durch die Gesichtsmerkmale (*fiducial points*), den Augen und Mundwinkeln im Schimpansengesicht, bestimmt wird. Sandwell u. a. [2013] nutzt die normalisierten Bilddaten der Arbeiten [Loos u. Ernst 2012, 2013] zusammen mit einem einheitlichen 3D Modell eines Schimpansengesichts nicht zur Normalisierung, sondern um künstlich größere Lerndatensätze zum Anlernen der Identifikationsalgorithmen zu generieren (siehe Kapitel 2.4.3).

Mit der Normalisierung z.B. von Gesichtern durch eine Affine oder Projektive Transformation auf den Bilddaten im zweidimensionalen Raum, wird das entsprechende Objekt als planar (Ebene im dreidimensionalen Raum) angenommen. In der Literatur konnte keine Arbeit gefunden werden, die eine Normalisierung von Gesichtern bei Tieren unter Berücksichtigung des Kopfes als Körper im dreidimensionalen Raum (nicht planar) behandelt (vgl. Kapitel 3.5).

2.4. Identifikation

2.4.1. Biometrische Identifikation beim Menschen

Die biometrische Identifikation (Erkennung, genaue Wiedererkennung) beschreibt das automatische Erkennen eines Individuums auf Grundlage von physiologischen oder verhaltensbasierten Merkmalen (nach Jain u. a. [2004]). Mit der biometrischen Identifikation ist es möglich die Identität eines Individuums sicherzustellen, wobei die individuellen Eigenschaften und nicht Besitz (z.B. Ausweis) oder individuelles Wissen (z.B. Passwort) ausschlaggebend sind (nach Jain u. a. [2004]). Grundsätzlich lassen sich die biometrischen Verfahren zur Identifikation (Abgleich eins-zu-mehreren) und zur Verifikation bzw. Authorisierung (Abgleich eins-zu-eins, z.B. biometrische Merkmale im Reisepass) einsetzen.

Jain u. a. [2004]; Delac u. Grgic [2004]; Goudelis u. a. [2008] und Adeoye [2010] stellen Übersichten von biometrischen Identifikationsverfahren beim Menschen zusammen. Die gängigsten biometrischen Merkmale, die beim Menschen eingesetzt werden, sind in Tabelle 2.1 aufgeführt.

Die zur Auswertung der Merkmale verwendeten Verfahren sind in der Tabelle 2.1 in die Kategorien „Bildverarbeitung“ und „Sonstige“ unterteilt. Die Kategorie „Bildverarbeitung“ beinhaltet dabei Methoden, welche anhand von Kamerabildern oder entsprechendem Bildmaterial (z.B. Fingerabdrucksensor) mit der Bildverarbeitung bewältigt werden. Bei den Verfahren zur Personenerkennung über Sprache werden häufig auch Methoden der Signal- und Bildverarbeitung eingesetzt, jedoch sind die Ausgangsdaten anderer Art.

In der Tabelle 2.1 sind Verweise zu Arbeiten des jeweiligen Themenbereichs aufgeführt, die jedoch nur beispielhaft sein sollen. Einige der biometrischen Verfahren beim Menschen oder vergleichbare werden bereits bei Tieren eingesetzt und sind in Kapitel 2.4.4 aufgeführt. Dabei ist für eine automatische Erkennung von Individuen bei Tieren darauf zu achten, dass eine Erkennung auch ohne eine Kooperation des Tieres durchführbar sein sollte. Bei den biometrischen Verfahren zur Erkennung von Menschen durch Fingerabdruck, Netzhaut oder Iris wird eine Kooperation der Person vorausgesetzt (nach Zhao u. a. [2003]). Systeme zur Gesichtserkennung (sowie der Infrarot Thermographie oder Gangarterkennung) können eine Person häufig auch ohne eine entsprechende Kooperation identifizieren (siehe [Zhao u. a. 2003]).

2. Kenntnisstand

Tabelle 2.1.: Übersicht biometrischer Identifikationsmerkmale beim Menschen

Bildverarbeitung	Sonstige
<ul style="list-style-type: none">• Infrarot Thermographie von Gesicht [Kong u. a. 2005] oder Hand bzw. der Venen [Khan u. a. 2009]• Gesichtserkennung (siehe Kapitel 2.4.2)• Gangart [Gafurov 2007]• Geometrie der Hand [Kumar u. a. 2003]• Fingerabdruck [Maltoni u. a. 2009]• Handflächenabdruck [Kong u. a. 2009]• Netzhautscan oder Iriserkennung [Bowyer u. a. 2008]• Ohren [Abaza u. a. 2013]• Unterschriften [Kholmatov u. Yanikoglu 2005]	<ul style="list-style-type: none">• Tastenanschlag [Banerjee u. Woodard 2012]• Geruch [Larsson u. a. 2000; Wongchoosuk u. a. 2009]• Sprache [Reynolds 2002]• DNA (Deoxyribonucleic acid) [Soram u. Khomdram 2010]

2.4.2. Gesichtserkennung zur Identifikation von Menschen

Normalisierung von Grauwerten

Das Normalisieren der Grauwerte in mehreren Bildern vom Gesicht einer Person dient dazu die Helligkeitsverteilung des Gesichts in den Aufnahmen anzupassen und die Aufnahmen in einen vergleichbaren Zustand zu bringen (siehe Kapitel 2.3.1 bzw. [Li u. a. 2006]). Die Grauwertnormalisierung wird häufig der Normalisierung (vgl. Kapitel 2.3.1) zugeordnet. Einige der Identifikationsverfahren zur *Gesichtserkennung*, wie das LBP-Verfahren (LBP - *local binary pattern* [Ahonen u. a. 2004], siehe nächstes Unterkapitel), sind jedoch nicht auf eine Grauwertnormalisierung angewiesen. In dieser Arbeit wurde daher das Kapitel (der Grauwertnormalisierung) den Identifikationsverfahren und nicht der Posennormalisierung (aus Kapitel 2.3.1) zugeordnet.

Eine Zusammenstellung von Verfahren zur Grauwertnormalisierung findet sich in [del Solar u. Quinteros 2008]. Die Arbeiten werden durch del Solar u. Quinteros [2008] in drei Kategorien unterteilt: Die Gesichtsmodellierung, die Normalisierung und Vorverarbeitung, und die invariante Merkmalsextraktion. Zur Gesichtsmodellierung werden lineare Unterräume mit kleiner Dimension eingesetzt, die dazu genutzt werden die Bildvariation durch unterschiedliche Beleuchtungssituationen von menschlichen Gesichtern zu modellieren. Die zweite Kategorie der Normalisierung und Vorverarbeitung braucht in den meisten Fällen keine vorherigen Anlern- oder Modellierungsschritte. Die in dieser Arbeit verwendete Histogrammäqualisation (siehe Kapitel 3.6.1) gehört in diese Kategorie (Normalisierung und Vorverarbeitung) und zählt nach del Solar u. Quinteros [2008] zu den „Universalwerkzeugen“ unter den Grauwertnormalisierungen. Die dritte Kategorie umfasst die invariante Merkmalsextraktion zu denen die diskrete Kosinustransformation (DCT - *discrete cosine transformation*, siehe nächsten Abschnitt zu [Goel u. Nehra 2011]) und die LBP (*local binary pattern*), welche auch zur Identifikation (siehe Unterkapitel *Gesichtserkennung*) eingesetzt werden, zählen. In [del Solar u. Quinteros 2008] werden Verfahren zur Grauwertnormalisierung der zweiten Kategorie (Normalisierung und Vorverarbeitung), die ohne ein vorheriges Anlernen auskommen, in Kombination mit Verfahren der *Gesichtserkennung* (siehe nächstes Unterkapitel) auf Basis der Hauptkomponentenanalyse (PCA - *principal component analysis*) verglichen.

Goel u. Nehra [2011] stellt einen Vergleich von fünf Verfahren zur Normalisierung von Beleuchtungsinhomogenitäten auf. Unter diesen Verfahren sind eine Histogrammäqualisation, wie sie in Kapitel 3.6.1 verwendet wird, zwei Verfahren, welche die DCT (*discrete cosine transformation*) einsetzen, um niedrige Frequenzen im Bild zu reduzieren bzw. zu entfernen, ein HF (homomorphe Filter) Verfahren und ein Verfahren zur Gamma-Korrektur der Intensitäten im Bild (GIC - *gamma intensity correction*). Die Verfahren, welche die DCT einsetzen, schneiden unter den genannten Verfahren am besten ab. In der bereits im Kapitel 2.3.1 beschriebenen Arbeit von Li u. a. [2013] wird ein diskriminierender Farbraum (DCS - *discriminant color space*) eingesetzt, wodurch eine maximale Separierbarkeit der einzelnen Klassen erreicht werden soll. Ein entsprechender Ansatz wird in dieser Arbeit nicht umgesetzt, da neben den Farbbildern der Kinect auch Grauwertbilder verarbeitet werden (siehe Industriekameras in Kapitel 3.3).

Kahraman u. a. [2007] stellt eine Methode zur Normalisierung von Gesichtern mit variierender Pose und Beleuchtung vor. Dabei setzt Kahraman u. a. [2007] eine Detektion des Gesichts voraus, mit der bereits eine grobe Positionierung des Gesichts bestimmt wurde. Zur Beleuchtungskorrektur wird eine Methode zur Anpassung des Histogramms an das Histogramm eines Ausgangsgesichts (Gesichtsmodell) durchgeführt. Die beiden Gesichtshälften werden hierzu getrennt voneinander in ihrer Helligkeit normalisiert. Anschließend werden die beiden Transformationen der Histogramme je Gesichtshälfte durch eine Gewichtung in eine lokale, von der Position (links / rechts) im Gesicht abhängige, Histogrammtransformation überführen. Die

2. Kenntnisstand

beschriebene Beleuchtungskorrektur wird mit einem AAM (*active appearance model*) kombiniert, bei dem sich iterativ das zuvor angelernte Modell (eines Gesichts) in Abhängigkeit von Form und Textur (Grauwertintensität) an das zu untersuchende Gesicht im Bild anpasst. Ist das Modell auf das aktuelle Gesicht angepasst wird eine virtuelle Frontansicht des Gesichts (vgl. Kapitel 3.5.3) zur weiteren Identifikation erzeugt.

Das in dieser Arbeit verwendete Verfahren der Histogrammäqualisation (siehe Kapitel 3.6.1) zur Normalisierung der Grauwerte ist, im Vergleich zu den vorgestellten Arbeiten, ein einfaches Verfahren ohne Parameter. Zudem ist bei einigen der vorgestellten Verfahren (z.B. die Verfahren der Gesichtsmodellierung aus [del Solar u. Quinteros 2008]) zu beachten, dass die in dieser Arbeit untersuchten Pferde aufgrund unterschiedlichster Zeichnungen im Allgemeinen keine homogene Gesichtsfarbe, wie etwa der Mensch, haben. Zudem werden in den Arbeiten [del Solar u. Quinteros 2008] und [Goel u. Nehra 2011] deutlich ausgeprägtere Inhomogenitäten der Beleuchtung mit Schattenbildung untersucht, als sie in dieser Arbeit vorkommen, da bereits bei der Aufnahme (siehe Aufbau in Kapitel 3.2) auf eine gute Ausleuchtung der Tiere geachtet wurde.

Gesichtserkennung

Übersichten zu Verfahren der Gesichtserkennung zur Identifikation von menschlichen Personen finden sich in den Arbeiten von Lu u. a. [2003]; Zhao u. a. [2003]; Bowyer u. a. [2006]; Abate u. a. [2007]; Zhang u. Gao [2009]; Jafri u. Arabnia [2009] und Muruganantham u. Jebarajan [2012]. Die Verfahren lassen sich in 2D- und 3D-Verfahren (siehe Unterkapitel *Gesichtserkennung 2D-Verfahren* bzw. *Gesichtserkennung 3D-Verfahren und Mehrfachansichten*) unterteilen.

Gesichtserkennung 2D-Verfahren

Wie in den folgenden Absätzen werden die 2D-Verfahren in den meisten Arbeiten anhand ihrer Feature in globale und lokale Verfahren unterteilt.

Die ursprünglichen lokalen (merkmalsbasierten) Verfahren, wie in [Lu u. a. 2003] erwähnt, basieren darauf zunächst typische Gesichtsmerkmale (Bezugspunkte - *fiducial points*), wie die Augen, die Nase und den Mund, zu lokalisieren. Unterschiedlichste Abmessungen der Komponenten im Gesicht (Gesichtsmerkmale) werden bei diesen Verfahren dazu genutzt einen Merkmalsvektor zu erstellen (siehe Lu u. a. [2003]). Solche Verfahren, die die typischen Merkmale im menschlichen Gesicht (Gesichtsmerkmale) auswerten, sind nach Lu u. a. [2003] darauf angewiesen die entsprechenden Merkmale zuverlässig aufzufinden und zuzuordnen, so dass die geometrischen Verhältnisse der aufgefundenen Merkmale zueinander stimmen. Aktuellere lokale Verfahren (z.B. LBP *local binary pattern* [Ahonen u. a. 2004]) sind nicht darauf angewiesen die typischen Gesichtsmerkmale aufzufinden, sondern verwenden lokale Feature, die nicht mit den hauptsächlichen Gesichtsmerkmalen (wie Auge, Nase und Mund)

übereinstimmen müssen. So beeinflussen die Gesichtsmerkmale zwar die lokalen Feature, jedoch sind die lokalen Feature nicht auf die Gesichtsmerkmale beschränkt. Die lokalen Feature werden häufig auf dem gesamten Gesicht erfasst (einige der Verfahren behandeln nur die charakteristischsten Feature z.B. Elastische Bündelgraphen EBG - *elastic bunch graph*) und in einen globalen Kontext gebracht. Das Verfahren der elastischen Bündelgraphen (EBG - *elastic bunch graph* bzw. EBGM - *elastic bunch graph matching*) gehört zu den lokalen Verfahren und wird nach Zhao u. a. [2003] zur Gesichtsdetektion (vgl. Kapitel 2.2.2), der Geschlechtsklassifikation beim Menschen oder der allgemeinen Objekterkennung eingesetzt. Beim EBG-Verfahren [Wiskott u. a. 1999] werden so genannte „Jets“ eingesetzt. Diese enthalten die Filterantworten mehrerer Gabor Wavelets (Faltungskerne), die pro „Jet“ typische Gesichtsmerkmale repräsentieren. Die Gabor Wavelets sind dabei dem menschlichen visuellen Wahrnehmungssystem nachempfunden [Zhao u. a. 2003]. Ein „Jet“, welcher jeweils ein typisches Gesichtsmerkmal (z.B. Auge) repräsentiert, sollte die Variation des jeweiligen Merkmals zwischen unterschiedlichen Personen (männlich/weiblich, Alter, Bart etc.) und Posenvariationen (siehe Zhao u. a. [2003]) abbilden. Sind die einzelnen „Jets“ im Gesicht positioniert (Auffinden der Gesichtsmerkmale) wird zwischen den sich ergebenden Punkten ein Graph erstellt. Die Ähnlichkeit zwischen Gesichtern wird durch die Ähnlichkeit der „Jets“ und der Verformung des Graphen bestimmt. Das LBP-Verfahren (LBP - *local binary pattern*) von Ahonen u. a. [2004] gehört ebenfalls zu den lokalen Verfahren und extrahiert invariante lokale Feature. Dabei wird in [Ahonen u. a. 2004] der LBP-Operator nach Ojala u. a. [1996] zur Beschreibung der Textur eingesetzt, welcher nach Ahonen u. a. [2004] Beleuchtungsvariationen innerhalb der Gesichter kompensiert und normalisiert (vgl. Unterkapitel *Normalisierung von Grauwerten*). Eine Zusammenstellung von Variationen des LBP-Verfahrens und deren Anwendung zur Gesichtserkennung findet sich in [Huang u. a. 2011].

Die globalen Verfahren hingegen arbeiten auf dem gesamten Bild (meistens Rohbild [Zhao u. a. 2003]), um eine Gesichtserkennung ohne lokale Merkmale durchzuführen. Das gesamte Gesichtsbild wird hierzu kodiert und durch einen Punkt in einem mehrdimensionalen Gesichtsraum repräsentiert. Die globalen Verfahren sind nach Lu u. a. [2003] darauf angewiesen, dass die Gesichter in einheitlichen Positionen, Orientierungen (Position und Orientierung ergeben die Pose) und Größen ausgewertet werden. Die Methode der Eigengesichter (Turk u. Pentland [1991b, a] siehe Kapitel 3.6.2) ist ein solches, nach Lu u. a. [2003] weit verbreitetes, globales Verfahren, welches auf der Hauptkomponentenanalyse (PCA *principal component analysis*) bzw. Karhunen-Loève Transformation (entspricht PCA) basiert. Die Methode der Fisherfaces ist eine weitere globale Methode. Sie basiert auf der linearen Diskriminanzanalyse (LDA - *linear discriminant analysis*), im Kontext mit den Fisherfaces auch als FLD (*fisher linear discriminant*) bezeichnet (siehe Lu u. a. [2003]). Die lineare Diskriminanzanalyse berücksichtigt, entgegen der Hauptkomponentenanalyse (PCA), die Inner- und Zwischen-Klassen-Varianz, um den komprimierten Gesichtsraum auf die zur Unterscheidung relevanten Informationen anzupassen (siehe

2. Kenntnisstand

Lu u. a. [2003]). Die Hauptkomponentenanalyse (PCA) ist nach Lu u. a. [2003] nicht robust gegenüber Variationen der Orientierung, der Positionierung (zusammen Pose vgl. Kapitel 2.3) oder der Beleuchtung in den Bildern der zu identifizierenden Gesichter. Bei entsprechenden Variationen nimmt die Erkennungs- bzw. Identifikationsleistung der Methode (PCA) ab, da die Feature der Methode sehr glatt werden (siehe Lu u. a. [2003]). Die Erkennungsraten der Hauptkomponentenanalyse (PCA) leidet dabei stärker bei Posen- oder Größenänderungen der Gesichter (85% bzw. 64% nach Turk u. Pentland [1991b]), wobei eine unterschiedliche Beleuchtung nach Jafri u. Arabnia [2009] (bei entsprechender Normalisierung, siehe Unterkapitel *Normalisierung von Grauwerten*) recht gut kompensiert werden kann (Erkennungsrate bei 96% auf 2500 Bildern von 16 Personen). Die Methode der Eigengesichter ist nach Zhang u. Gao [2009] eine schnelle, einfache und praktische Methode, wodurch sie zu der am weitesten verbreiteten Methode zur Gesichtserkennung beim Menschen wurde.

Ruiz-del Solar u. Navarrete [2005] vergleicht mehrere „standard“ Projektionsmethoden, darunter die Hauptkomponentenanalyse (PCA - *principal component analysis*), die lineare Diskriminanzanalyse nach Fisher (FLD - *fisher linear discriminant*) sowie ein evolutionäres Verfahren (EP - *evolutionary pursuit*) mit unterschiedlichen Distanz- und Ähnlichkeitsmaßen (euklidische Distanz, Mahalanobis Distanz, Kosinusdistanz und weitere). Zudem werden in Ruiz-del Solar u. Navarrete [2005] nichtlineare Kernel-Projektionen (Kernel PCA, KPCA - *kernel principal component analysis*) und differentielle Verfahren verglichen. Nach Ruiz-del Solar u. Navarrete [2005] bietet die lineare Diskriminanzanalyse nach Fisher (FLD - *fisher linear discriminant*) theoretisch die bessere Unterscheidungsfähigkeit gegenüber der Hauptkomponentenanalyse (PCA - *principal component analysis*). Wird jedoch mit einer kleineren Anzahl von Gesichtern oder mit atypischen Gesichtern angelernt, so zeigt die Hauptkomponentenanalyse (PCA - *principal component analysis*) gegenüber der linearen Diskriminanzanalyse nach Fisher (FLD - *fisher linear discriminant*) nach Ruiz-del Solar u. Navarrete [2005] eine bessere Generalisierbarkeit (robuster gegenüber veränderten Daten).

Im Vergleich zu den globalen Verfahren, sind die lokalen (merkmalsbasierten) Verfahren nach Zhao u. a. [2003] weniger anfällig gegenüber Beleuchtungsvariationen, unterschiedlichen Betrachtungswinkeln und ungenauer Gesichtspositionierung (Position der Gesichter, vgl. Kapitel 2.3.1).

Neben der bisherigen Unterteilung anhand der Feature lassen sich die Methoden auch in lineare und nicht-lineare Verfahren unterteilen (z.B. in Abate u. a. [2007] oder Muruganantham u. Jebarajan [2012]). In Abate u. a. [2007] wird zu den genannten Verfahren (PCA, FLD bzw. LDA) die Methode der DCV (*discriminant common vectors*) und die unabhängige Komponentenanalyse (ICA - *independent component analysis*) als lineares Verfahren erwähnt. Die linearen Verfahren verwenden nach Abate u. a. [2007] lediglich das Amplitudenspektrum der Bilder, wobei

das Phasenspektrum nicht ausgewertet wird. Zu den nicht-linearen Verfahren aus Abate u. a. [2007] gehören die Kernel PCA (KPCA - *kernel principal component analysis*), die Neuronale-Netze (vgl. tiefe Neuronale Netze - DNN (*deep neural networks*) [Taigman u. a. 2014]), die isometrische Transformation (*isometric transformation*, ISOMAP [Tenenbaum u. a. 2000]) sowie die selbstorganisierenden Karten (SOM - *self-organizing map*). Die unabhängige Komponentenanalyse (ICA - *independent component analysis*) und die Kernel PCA (KPCA - *kernel principal component analysis*) sind Generalisierungen der Hauptkomponentenanalyse (PCA).

In Abbildung 2.1 sind die hauptsächlichen Verfahren nochmals dargestellt. Die ursprüngliche Abbildung stammt von Muruganantham u. Jebarajan [2012], in deren Arbeit die Verfahren in vier Hauptkategorien unterteilt werden: Schablonenabgleich (*template matching*), Neuronale Netze (*neural networks*), Model- (*model based*) und Aussehenbasiert (*appearance based*). Die modellbasierten Verfahren werden weiter in 2D- und 3D-Verfahren unterteilt. Zu den 2D-Verfahren zählt die Methode der elastischen Bündelgraphen (EBG, siehe oben) sowie die aktiven Erscheinungsmodelle (*active appearance model*).

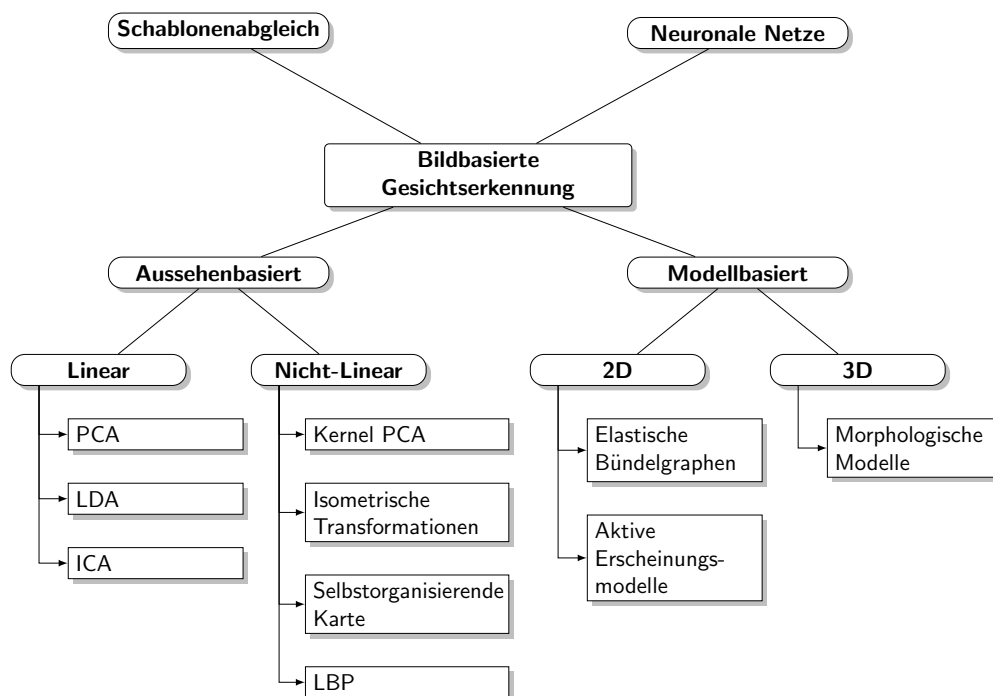


Abbildung 2.1.: Übersicht der Identifikationsverfahren im 2D [Verändert nach: Muruganantham u. Jebarajan 2012]

2. Kenntnisstand

Unter den 3D-Verfahren sind morphologische Modelle (*3D morphological model*) aufgeführt (z.B. [Blanz 2006]). Die aussehensbasierten Verfahren werden in der Abbildung 2.1 (bzw. durch Muruganantham u. Jebarajan [2012]) in lineare (PCA, LDA und ICA) und nicht-lineare Verfahren (Kernel PCA - KPCA, selbstorganisierenden Karten - SOM und isometrische Transformationen ISOMAP) unterteilt. Das LBP-Verfahren wurde in Abbildung 2.1 ergänzt (vgl. Abbildung aus [Muruganantham u. Jebarajan 2012]).

Teilweise wird zudem zwischen der Erkennung in Einzelbildern und der in Videosequenzen unterschieden (z.B. Jafri u. Arabnia [2009]), in denen ein Tracking möglich wird. Jedoch lässt sich eine entsprechende Unterscheidung eher auf die Detektion der Gesichter (siehe Kapitel 2.2.2) beziehen. Li u. a. [2013] stellt weiter einen Klassifikator vor, der eine spärliche Repräsentation (SRC - *sparse representation classifier*) von Wright u. a. [2009] nutzt.

Gesichtserkennung 3D-Verfahren und Mehrfachansichten

Wie bereits erwähnt sind gerade globale Verfahren (z.B. LDA und PCA) sehr sensitiv gegenüber Posenvariationen, da die globalen Ansätze Gesichter auf Grundlage der Grauwerte in den einzelnen Pixeln im Bild unterscheiden. Die Rotation des menschlichen Gesichts (als 3D Objekt) führt jedoch nahezu immer zu einem Pixelversatz in den 2D Bildern [Zhang u. Gao 2009]. Nach Abate u. a. [2007] kann entsprechenden Posenvariationen mit einer Posennormalisierung (siehe Kapitel 2.3) begegnet werden oder aber indem für die unterschiedlichen Posen entsprechende Identifikationsmethoden angelernt werden (Gesichtserkennung aus Mehrfachansichten - *multi-view face recognition*). Abate u. a. [2007] stellt mehrere Verfahren zur Gesichtserkennung mit 3D Daten (bzw. auch nur 2,5 D Tiefendaten) vor, wobei viele der genannten Verfahren ähnlich der Verfahren aus Kapitel 2.3 eine Posenkorrektur auf Basis der 3D Daten durchführen und die eigentliche Erkennung im 2D durchführen, wie auch in dieser Arbeit. Eine Alternative stellen die in [Abate u. a. 2007] zusammengetragenen Methoden dar, die die Oberflächenkrümmung (*curvature curves*) der Gesichter nutzen um Kontrollpunkte (Bezugspunkte - *fiducial points*) im menschlichen Gesicht aufzufinden (entspricht Landmarken vgl. Kapitel 3.5.2) und anhand dieser Kontrollpunkte eine Normalisierung durchführen (siehe Kapitel 2.3). Ferner lässt sich die Oberflächenkrümmung (z.B. globale und lokale Krümmung bzw. von der Oberflächenkrümmung abhängige Punktsignaturen) direkt zur Unterscheidung der Gesichtsformen beim Menschen einsetzen (nach Abate u. a. [2007]). Bowyer u. a. [2006] und Abate u. a. [2007] stellen weitere 3D Identifikationsverfahren zusammen, wobei einige die bereits beschriebenen Methoden der Identifikation aus 2D Gesichtsbildern auf 3D Daten anwenden (z.B. PCA) oder beide Daten (2D+3D, 4D bestehend aus Textur und 3D) kombinieren. In dieser Arbeit wird die Erkennung lediglich auf den 2D Daten durchgeführt, weshalb aus Gründen des Umfangs nicht auf weitere 3D Identifikationsmethoden eingegangen wird.

2.4.3. Bildverarbeitungsmethoden zur Identifikation von Wildtieren

Kühl u. Burghardt [2013] stellt eine Übersicht von biometrischen Tiermerkmalen und einigen Methoden zusammen. Zusammengetragen sind außerdem die Vorzüge einer hohen Objektivität, Vergleichbarkeit und Wiederholbarkeit, die die automatische Auswertung der biometrischen Merkmale bei Tieren für Studien des Verhaltens oder der Population mit sich bringen.

Burghardt [2008] beschreibt in seiner Arbeit die Identifikation von Pinguinen [Burghardt u. a. 2004b], Zebras und weiteren Tieren anhand der individuellen Fellzeichnungen. Kelly [2001] nutzt die Fellzeichnung von Geparden um zwischen einzelnen Tieren zu unterscheiden. Hoque u. a. [2011] beschreibt eine Methode zur Unterscheidung von Kammolchen (Salamanderart) anhand ihrer einzigartigen Bauchzeichnung. Gamble u. a. [2008] unterscheidet Marmor-Querzahnmolche anhand der Zeichnungen auf dem Rücken der Schwanzlurchart. Lahiri u. a. [2011] zeigt eine weitere Methode zur Unterscheidung von Zebras und Tieren mit entsprechend ausgeprägter Fellzeichnung.

Araabi u. a. [2000]; Markowitz u. a. [2003]; Gilman u. a. [2013] erkennen Delphine anhand der Form ihrer Rückenflosse. Hughes u. a. [2013] nutzt Bilder der Rückenflossen von Weißen Haien, um die einzelnen Tiere zu identifizieren. Rangelova u. a. [2004] erkennt Individuen bei Buckelwalen durch die unterschiedliche Pigmentierung der Schwanzflosse. Van Tienhoven u. a. [2007] stellt eine Software zur Unterscheidung von Sandhaien anhand der individuellen Punktmuster an den Seiten der Tiere vor. Hillman u. a. [2003] beschreibt ein Verfahren zur Unterscheidung mehrerer Delphin- und Walarten. Shrivakshan u. Chandrasekar [2011] hingegen beschreiben eine Methode mit der unterschiedliche Haiarten unterschieden werden können. Town u. a. [2013] erkennt anhand individueller Zeichnungen am Bauch von Mantarochen, um welches Tier es sich handelt.

Die Identifikation einzelner Individuen bei Tieren wird durch Biologen und Wissenschaftler dazu genutzt, das Verhalten und soziale Strukturen zwischen den Tieren zu studieren sowie die Größe und Populationsverteilungen von Tierarten bestimmen zu können. Zur Identifikation werden häufig immer noch Marker an den Tieren angebracht.

Redd u. a. [2012] beschreibt ein Verfahren zur Unterscheidung von Papageien anhand von Aufnahmen der oberen Schnabelspitze. Anderson u. a. [2010] identifiziert Individuen bei Eisbären anhand des Musters der Barthaare. Ardovini u. a. [2007, 2008] erkennt einzelne Tiere bei Elefanten durch die Form der Ohren. Hiby u. a. [2009] nutzt ein 3D Modell zur Erkennung von Tigern anhand ihrer Streifenmuster, wobei die Orientierung der Kamera und die Pose der Tiere für einen Ähnlichkeitsvergleich zur Identifikation des Individuums berücksichtigt wird.

Die bisher vorgestellten Methoden (mit Ausnahme von [Burghardt 2008]) sind darauf angewiesen, dass ein Benutzer auf den Bildern die Tiere oder die entsprechenden Körperteile markiert (keine *Detektion* vgl. Kapitel 2.2 bzw. Landmarkendetektion

2. Kenntnisstand

vgl. Kapitel 2.3.2). In der Arbeit von Burghardt u. Campbell [2007]; Burghardt [2008] und Sherley u. a. [2010] werden ebenfalls 3D Modelle eingesetzt, um eine Identifikation von Pinguinen zu ermöglichen.

Gesichter

Dabarera u. Rodrigo [2010] beschreibt die Identifikation von Elefanten anhand von Frontalaufnahmen der Gesichter und verwendet, wie auch in Kapitel 3.6.2, die Methode der Eigengesichter (PCA - *principal component analysis*). Loos u. Ernst [2013] stellt ein gesamtes Framework zur automatischen Identifikation von Schimpansen vor. Die Detektion wurde bereits in Kapitel 2.2.3 beschrieben, die Normalisierung in Kapitel 2.3. In der Arbeit Loos u. Ernst [2013] wird ein globales Verfahren, welches Gabor Features verwendet, mit einem lokalen, welches SURF (siehe Kapitel 2.2.1 bzw. [Bay u. a. 2008]) Features nutzt, zur Identifikation der Schimpansengesichter kombiniert. In [Loos u. a. 2011a; Loos u. Ernst 2012] wird zur Identifikation von Schimpansen- und Gorillagesichtern der Algorithmus von Wright u. a. [2009] eingesetzt. In der Arbeit Loos u. a. [2011b] werden mehrere Verfahren der menschlichen Gesichtserkennung (siehe Kapitel 2.4.2) ohne besondere Anpassungen auf Schimpansen- und Gorillagesichtern verglichen. In [Loos 2012] wird das ursprüngliche Verfahren [Wright u. a. 2009] erweitert und in [Loos u. Pfitzer 2012] wird eine Normalisierung der Gesichter nach Huang u. a. [2007] (siehe Kapitel 2.3.2) mit einer hybriden Methode aus globalen und lokalen Features genutzt. Eine Fusion aus lokalen und globalen Features wird auch in der Arbeit von Loos [2013] behandelt.

Die Arbeiten [Loos u. a. 2011a, b; Loos 2012; Loos u. Ernst 2012; Loos u. Pfitzer 2012; Loos u. Ernst 2013] und [Loos 2013] beschreiben den Einsatz von Verfahren der menschlichen Gesichtserkennung bei Primaten. Die Gesichter der Primaten weisen, im Vergleich zu anderen Tierarten, große Ähnlichkeiten mit den Gesichtern des Menschen auf (vergleichsweise homogene Gesichtsfärbung, Position der Gesichtsmarkmale, etc.). Sandwell u. a. [2013] nutzt ein generalisiertes 3D Modell der Schimpansengesichter und erzeugt künstliche Lerndaten, mit denen die Gesichtserkennung angelernt und damit auf den Schimpansengesichtern (auch auf realen Bilddaten) merkbar robuster wird.

In [Kumar u. Singh 2014] werden mehrere lokale (z.B. LBP, siehe Kapitel 2.4.2) bzw. globale Verfahren (z.B. PCA, LDA und ICA, siehe Kapitel 2.4.2) der menschlichen Gesichtserkennung auf Hundegesichter angewendet und ausgewertet. Mit dem lokalen Verfahren LBP (*local binary pattern* [Ahonen u. a. 2004]) wird die beste Erkennungsrate von 81,15% (Rang-1, das zu ermittelnde Tier ist an erster Stelle) bzw. 93,86% (Rang-4, die korrekte Klassifikation des individuellen Tieres findet sich an vierter Stelle) unter den verglichenden Verfahren erreicht.

2.4.4. Identifikationsmethoden von landwirtschaftlichen Nutztieren

Coulon u. a. [2009] zeigt, dass Rinder (bzw. Färsen; Jungkuh, ohne Kalb) der „Holstein“ Rasse in der Lage sind Individuen anhand von 2D Bildern ihrer eigenen Rasse, aber auch anderer Rinderrassen zu unterscheiden. Kendrick u. Feng [2011] beschreibt, dass auch Schafe diese, für ein soziales Zusammenleben so wichtige Fähigkeit, besitzen. Zudem sind Schafe nach Kendrick u. Feng [2011] in der Lage unterschiedliche Gemütszustände (hauptsächlich durch die Stellung der Ohren des Gegenüber) von Gesichtern auf Bildern auszumachen.

Marchant [2002] beschreibt, wie wichtig es ist die Identität einzelner Tiere in Landwirtschaftsbetrieben verifizieren zu können, da dies zur Lebensmittelsicherheit und deren Qualitätssicherung beiträgt, sich Krankheitsausbrüche zurückverfolgen und damit eindämmen lassen, Kosten durch Regierungskontrollen (Intervention und Ausrottung) reduziert werden und mögliche Verluste beim Handel minimiert werden. Dabei geht es nach Marchant [2002] um die Gesundheit der Tiere, die durch kontaminierte Fleischprodukte Einfluss auf die Gesundheit der Konsumenten haben kann. Die gesicherte Identität und Zurückverfolgbarkeit der Tiere dient damit dem Einschränken von Epidemien, welches zugleich die Risiken zur Übertragung von Krankheiten auf den Menschen minimiert. In [Marchant 2002] werden die unterschiedlichen Verifikationsmethoden wie das Brandmarken, das Setzen von Schwanzmarken oder Ohrmarken (Metall, Plastik, Elektronisch), das Implantieren von Chips (z.B. RFID *radio-frequency identification*) oder biometrische Verfahren, wie Schnauzen- bzw. Nasenabdruck, Iris-Scan und Aufnahmen der Retina, aufgeführt. DNA Profile können zum Zurückverfolgen von Fleischprodukten zu einem bestimmten Tier genutzt werden. Nach Marchant [2002] werden vorrangig nicht-biometrische Verfahren genutzt, wobei die Ohrmarken am weitesten verbreitet sind. Zudem sind gekapselte RFID-Transponder zum Schlucken (Bolus) eine weitere Form der eingesetzten elektronischen Transponder (z.B. [Rojas-Olivares u. a. 2012]).

Barron u. a. [2009] und Shanahan u. a. [2010] stellen bestehende Probleme beim Verwenden von Ohrmarken dar: Die Lesbarkeit der Ohrmarken ist nicht gegeben, Ohrmarken gehen verloren, gesundheitliche Probleme durch Ohrmarken treten auf und eine Sabotage ist leicht möglich (Austausch der Ohrmarken von unterschiedlichen Tieren). Daher wird durch Shanahan u. a. [2010] (Rinder) und Barron u. a. [2009] neben dem Verwenden von elektronischen Ohrmarken mit RFID (*radio-frequency identification*) eine zusätzliche Sicherung durch biometrische Merkmale der Tiere vorgeschlagen, um die genannten Nachteile beim Einsatz der Ohrmarken zu reduzieren. In [Barron u. a. 2009; Shanahan u. a. 2010] werden als biometrisches Verfahren zur sicheren Identitätsbestimmung der Tiere Aufnahmen der Retina vorgeschlagen. In [Barron u. a. 2009] wird neben den DNA Profilen und Schnauzenabdrücken, die ebenfalls in [Shanahan u. a. 2010] erwähnt werden, auch die Gesichtserkennung als biometrisches Merkmal zur Identitätsbestimmung einzelner landwirtschaftlicher Nutztiere (siehe [Corkery u. a. 2007] in Kapitel 2.4.4) angesprochen. Nach Barron u. a. [2009] ist die Gesichtserkennung auf Kamerabildern (bzw. die Aufnahme von

2. Kenntnisstand

Schnauzenabdrücken) einer der nächsten Schritte, um ein vollautomatisches System zur Identifikation (Identifikation neben Verifikation denkbar) von Nutztieren aufbauen zu können. In [Barron u. a. 2009] wird dazu ein passendes Aufnahmesystem (vgl. Kapitel 3.3) sowie eine Software, um gute Bilder (vgl. Kapitel 2.2 und 2.3) zur Identifikation (vgl. Kapitel 2.4.2) aufzunehmen, benötigt.

Die unterschiedlichen Markierungsmethoden zeigen verschiedene Verträglichkeiten bei den Tieren. In [Schatzmann 2012] werden dazu die Kennzeichnungsmethoden des Heißbrands und der Transponder-Implantation bei Pferden verglichen. Nach Schatzmann [2012] wurden nach einer Kennzeichnung durch Heißbrand keine sonderbaren Veränderungen des Gewebes bei den untersuchten Pferden festgestellt, die auf Spätfolgen oder chronische Entzündungen bei den Tieren hinweisen. Die untersuchten Pferde, welche mittels Transponder-Implantaten gekennzeichnet wurden, zeigten hingegen ausgeprägte entzündliche Veränderungen der Haut, welche auch zu Spätfolgen und länger andauernden Schmerzen bei den Tieren führen können. Um die Pferde zu kennzeichnen ist nach Schatzmann [2012] die Methode des Heißbrands der Transponder-Implantation vorzuziehen, um „Leidensfreiheit bzw. Abwesenheit von Schmerzen und uneingeschränkte Bewegungsfreiheit,“ (siehe [Schatzmann 2012]) hinsichtlich einer artgerechten Haltung zu gewährleisten.

Bei den genannten Systemen der Rückverfolgbarkeit reicht eine Verifikation der Identität eines Tieres. Bei der Präzisionstierhaltung (PLF - *precision livestock farming*) geht es unter anderem um das individuelle Tierwohl [Berckmans 2006; Kashiha u. a. 2013]. So dass neben einer Verifikation die Identifikation eines Tieres (Abgleich ein-zu-mehreren) nötig ist, um Verhalten und Erkranken eines einzelnen Tieres erfassen zu können.

Stankovski u. a. [2012] beschreibt hierzu ein System zum Monitoring der Melkzyklen bei Milchkühen, welches mittels der eingesetzten RFID-Transponder (*radio-frequency identification*, hier im UHF Ultrahochfrequenz *ultra high frequency* Bereich von 915 MHz) eine durchschnittliche Erkennungsgenauigkeit von 99,8 % erreicht. In [Porto u. a. 2014] werden aktive RFID-Transponder (im UWB Ultraweitband *ultra wide band*) eingesetzt, um neben der Identifikation eine Lokalisierung von Milchkühen im Stall zu ermöglichen. Porto u. a. [2014] stellt fest, dass ein Kompromiss zwischen Identifikations- und Lokalisationsleistung mit den eingesetzten Transpondern zu treffen ist und empfiehlt eine 98 %ige Identifikation mit einer Genauigkeit von $0,52 \pm 0,36 m$ bei der Positionsbestimmung zu kombinieren. Eine Übersicht von unterschiedlichen RFID-Transpondern (*radio-frequency identification transponder*) findet sich in [Weis 2007].

Identifikationsmethoden der Bildverarbeitung

Anguzza [2013] nutzt bei Kühen Markierungen mit hautverträglicher Farbe, um diese in den Bilddaten aufzufinden und die aufgenommenen Individuen anhand der Markierungen zu identifizieren. Kashiha u. a. [2013] zeigt eine vergleichbare Kennzeichnung (unterschiedliche Muster werden auf jedem Tier appliziert) zur Identifikation von Schweinen.

Kim u. a. [2005a] schlägt eine Identifikation von Rindern der Rasse „Holstein“ anhand der unterschiedlichen Fellzeichnungen vor. Hierzu wird zunächst in seitlichen Aufnahmen das Tier vom Hintergrund (in [Kim u. a. 2005a] einfarbig) separiert, um mit einem Schwellwertverfahren ein Binärbild der Fellzeichnung zu generieren. Die Binärwertbilder der einzelnen Tiere werden anschließend durch ein zuvor trainiertes Neuronales-Netz unterschieden. Das Neuronale-Netz wurde in [Kim u. a. 2005a] auf Bildern von 49 Holstein Kühen angelern und anschließend auf 10 Bildern getestet, ohne eine Identifikationsgenauigkeit anzugeben.

In den Arbeiten [Barron u. a. 2009; Shanahan u. a. 2010] wird die Aufnahme der Retina zur Verifikation der Identität bei Rindern vorgeschlagen. Die Arbeiten von Barron u. a. [2008]; Barry u. a. [2008]; Rojas-Olivares u. a. [2011] und Rojas-Olivares u. a. [2012] befassen sich mit der Auswertung der Retina-Aufnahmen bei Schafen bzw. Lämmern.

Suzaki u. a. [2001] beschreibt die Identitätsfeststellung von Pferden durch Aufnahmen der Iris. Ein vergleichbares System zur Verifikation der Identität von Tieren anhand von Irisaufnahmen wird in [Musgrave u. L. 2002] beschrieben. Zhao u. a. [2011] schlägt vor, die Iriserkennung von Rindern, Schweinen und anderen großen Tieren der Landwirtschaft in einem zweidimensionalen Code zu kodieren, um die Zurückverfolgbarkeit der Tiere und deren Fleischprodukte zu gewährleisten.

Petersen [1922]; Hirsch u. a. [1952] zeigen, dass eine Unterscheidung einzelner Individuen bei Rindern durch einen Nasenabdruck möglich ist. In den Arbeiten von Minagawa u. a. [2002]; Barry u. a. [2007]; Noviyanto u. Arymurthy [2012, 2013] und Awad u. a. [2013b, a] werden Verfahren der Bildverarbeitung angewendet, um entweder auf Papier gebrachte Nasenabdrücke (z.B. [Minagawa u. a. 2002; Noviyanto u. Arymurthy 2013]) zu vergleichen oder solche die mit einer Kamera (z.B. Noviyanto u. Arymurthy [2012]; Awad u. a. [2013b]) aufgenommen wurden.

Gesichter

Die Gesichtserkennung von landwirtschaftlichen Nutztieren (bzw. Identifikation anhand von Kopfaufnahmen) ist in der Literatur deutlich seltener vertreten als z.B. die von anderen Tieren bzw. Wildtieren aus Kapitel 2.4.3. Trotz sorgfältiger Recherche, konnten lediglich drei in dieses Unterkapitel passende Arbeiten gefunden werden.

Kim u. a. [2005b] führt die Identifikation anhand von Aufnahmen der Köpfe von zwölf Japanischen Rindern mit einem assoziativen Speichernetzwerk (*associative memory*, Unterart der Neuronalen-Netze [Specht 1988, 1990]) durch. Es wird lediglich ein Originalbild pro Tier angelernt, welches anschließend transformiert wird (rotiert, translatiert etc.), um zu testen, ob eine Erkennung weiterhin möglich ist. Kim u. a. [2005b] kommt zu dem Schluss, dass der vorgestellte Algorithmus nicht zur Identifikation im Stall geeignet ist. Obwohl der Algorithmus auf Bildern mit Helligkeitsschwankungen, Verzerrung und Rauschen die Tiere wiedererkennt, schlägt die Identifikation nach einer Rotation der Bilder mit dem in [Kim u. a. 2005b] vorgestellten Neuronalen-Netz fehl.

Corkery u. a. [2007] zeigt eine Gesichtserkennung auf einer Gruppe von 50 Schafen. Corkery u. a. [2007] kombiniert hierzu die Methode der Eigengesichter (Hauptkomponentenanalyse, PCA *principal component analysis* siehe Kapitel 3.6.2) mit der unabhängige Komponentenanalyse (ICA, siehe Kapitel 2.4.2). Die Unterscheidung der Tiere geschieht über die Kosinus-Distanz im „Gesichtsraum“ (Unterraum vgl. Kapitel 3.6.2). Corkery u. a. [2007] gibt Erkennungsraten von 95,3 % bis 96 %, bei einer Falsch-Positiv Rate von 4 % an. Nach Corkery u. a. [2007] haben Schafe im Vergleich zum Menschen eine deutlich reduzierte Mimik (Gesichtsausdrücke). Dies führt bei Schafen zu besseren Identifikationsergebnissen als beim Menschen, wenn Verfahren der Gesichtserkennung eingesetzt werden, die anfällig auf Variationen durch unterschiedliche Gesichtsausdrücke sind (siehe [Corkery u. a. 2007]). Cai u. Li [2013] vergleicht mehrere Verfahren der Gesichtserkennung mittels LBP-Operator (LBP - *local binary pattern*, Abwandlungen von [Ahonen u. a. 2004], siehe Kapitel 2.4.2) auf einer Gruppe von 30 Rindern. Cai u. Li [2013] gibt eine Erkennungsrate von bis zu 95,3 % beim Training auf 90 % (90 Bilder pro Tier) der Daten an.

3. Material und Methoden

Dieses Kapitel beschreibt sowohl die Materialien als auch die bestehenden und entwickelten Methoden für die in dieser Arbeit umgesetzte optische Identifikation einzelner Individuen bei Pferden und deren Evaluation. Zunächst wird in Abschnitt 3.1 auf die Tiere eingegangen, welche im Zuge der Versuchsreihen untersucht wurden. Im Unterkapitel 3.2 wird die Anordnung der im Versuchsstall aufgenommenen Objekte (Pferdeköpfe der Tiere aus Abschn. 3.1) zu den Geräten aus Abschnitt 3.3 beschrieben. Die Geräte aus Abschnitt 3.3 wurden dazu genutzt, die optischen Daten zu den Versuchsreihen aufzuzeichnen. In den folgenden Abschnitten 3.4 „*Detektion*“, 3.5 „*Posenbestimmung und Normalisierung*“ und 3.6 „*Identifikation der einzelnen Individuen bei Pferden*“ werden die Methoden der zur Identifikation eingesetzten Bildverarbeitungskette aus Abbildung 1.1 aufgeführt und erläutert. Zuletzt werden in den Abschnitten 3.7 und 3.8 Angaben zur Versuchsdurchführung, sowie zur Aufbereitung und Auswertung der Daten gemacht.



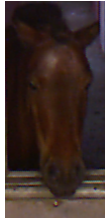

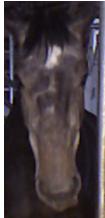
3.1. Eingesetzte Tiere



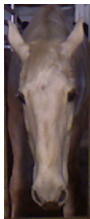


Der Großteil (bis auf „Baldo“ und „Royal Bess“) der in der Übersicht 3.1 abgebildeten Pferde lebt in einem Herdenverband von bis zu 15 Tieren. Die Tiere werden saisonal von etwa November bis März in dem unter Abschnitt 3.2 näher beschriebenen Offenstall gehalten, in dem sie sich frei bewegen können. Die restliche Zeit verbringen die Tiere auf den Weiden, wo keine weiteren Aufnahmen erstellt wurden. Die Pferde „Baldo“ und „Royal Bess“ wurden lediglich kurze Zeit im Offenstall gehalten und sind die gesamte restliche Zeit auf der Weide.

Die Abbildung 3.1 enthält neben dem Namen, dem Alter (im Januar 2013) und der Rasse der untersuchten Tiere jeweils eine Frontaufnahme des Kopfes. In Tabelle 3.1 ist eine Zusammenstellung der Fellfarben und Zeichnungen der Pferde zu finden. Zur Untersuchung standen mehrere gleichfarbige (z. B. Anna, Diadem und Romanze) sowie farblich recht unterschiedliche (z. B. Serenade und Rosi) Tierköpfe zur Verfügung, wobei einige Tiere ähnliche Zeichnungen aufweisen.

Eine Besonderheit sind die Tiere „Bara“ und „Rosi“, welche Geschwister sind. Sie zeichnen sich durch eine für den Beobachter unterscheidbare, jedoch sehr ähnliche Fellfarbe wie auch Zeichnung aus. Wohingegen die Tiere „Anna“ und „Blüte“, die ebenfalls Geschwister sind, deutlich weniger Ähnlichkeit zueinander aufweisen. Weitere Details zu den Tieren sind im Anhang A aus den Pferdepässen zusammengetragen.

3. Material und Methoden

Name	Anna	Arielle	Baldo	Bara	Blüte
Alter	4 Jahre	4 Jahre	18 Jahre	3 Jahre	3 Jahre
Rasse	Holsteiner	Holsteiner	Shetland Pony	Holsteiner	Holsteiner
					

Name	Blütenzauber	Delia	Diadem	Dumbledore	Linus
Alter	3 Jahre	1 Jahr	24 Jahre	1 Jahr	1 Jahr
Rasse	Holsteiner	Holsteiner	Holsteiner	Holsteiner	Holsteiner
					

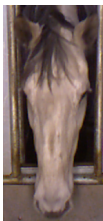

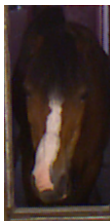

Name	Romanze	Rosi	Royal Bess	Serenade
Alter	11 Jahre	6 Jahre	13 Jahre	16 Jahre
Rasse	Holsteiner	Holsteiner	Welsh A (Pony)	Welsh D (Pony)
				

Abbildung 3.1.: Pferde Übersicht

3.1. Eingesetzte Tiere

Tabelle 3.1.: Fellfarbe und Zeichnungen der Tiere

Name	Rasse	Fellfarbe	Zeichnung
Anna	Holsteiner	Schimmel	leichte Schnippe
Arielle	Holsteiner	Dunkelbrauner	Blume
Baldo	Shetland Pony	Brauner	-
Bara	Holsteiner	Brauner	schmale Blesse
Blüte	Holsteiner	Schimmel	Flämmchen
Blütenzauber	Holsteiner	Braunschimmel	unregelmäßige Blesse
Delia	Holsteiner	Brauner	schmale Blesse
Diadem	Holsteiner	Schimmel (Braunschimmel)	-
Dumbledore	Holsteiner	Schimmel	leichte Blume
Linus	Holsteiner	Schimmel	schmale Blesse
Romanze	Holsteiner	Schimmel	-
Rosi	Holsteiner	Fuchs	breite Blesse
Royal Bess	Welsh A (Pony)	Brauner	breite Blesse
Serenade	Welsh D (Pony)	Rappe	Blume

3.2. Versuchsstall / Messanordnung

Die Versuche wurden im Offenstall der Firma HIT (Hinrichs Innovation + Technik GmbH) durchgeführt. Ein Grundriss des Offenstalls ist in Abbildung 3.2 gezeigt. Die Tiere (siehe Unterkapitel 3.1) werden hier in einer Gruppe freilaufend gehalten und haben Zugang zu allen nicht grau markierten Flächen. Es gibt einen Bereich, der mit Sand ausgestreut ist, sowie einige Bereiche, die einen befestigten Untergrund haben. Die Bereiche mit festem Untergrund teilen sich in einen Bereich, der überdacht ist und einen nicht überdachten Bereich (Außenbereich, hellgrün) auf. Zudem haben die Tiere Zugang zum Inneren des Gebäudes (Innenbereich, dunkelgrün), dessen Tore ständig geöffnet sind, so dass die Tiere sich jederzeit zwischen diesen Bereichen bewegen können. Der Innenbereich hat eine Fläche von ca. 124 m^2 und der Außenbereich von ca. 910 m^2 , wovon ca. 88 m^2 überdacht sind.

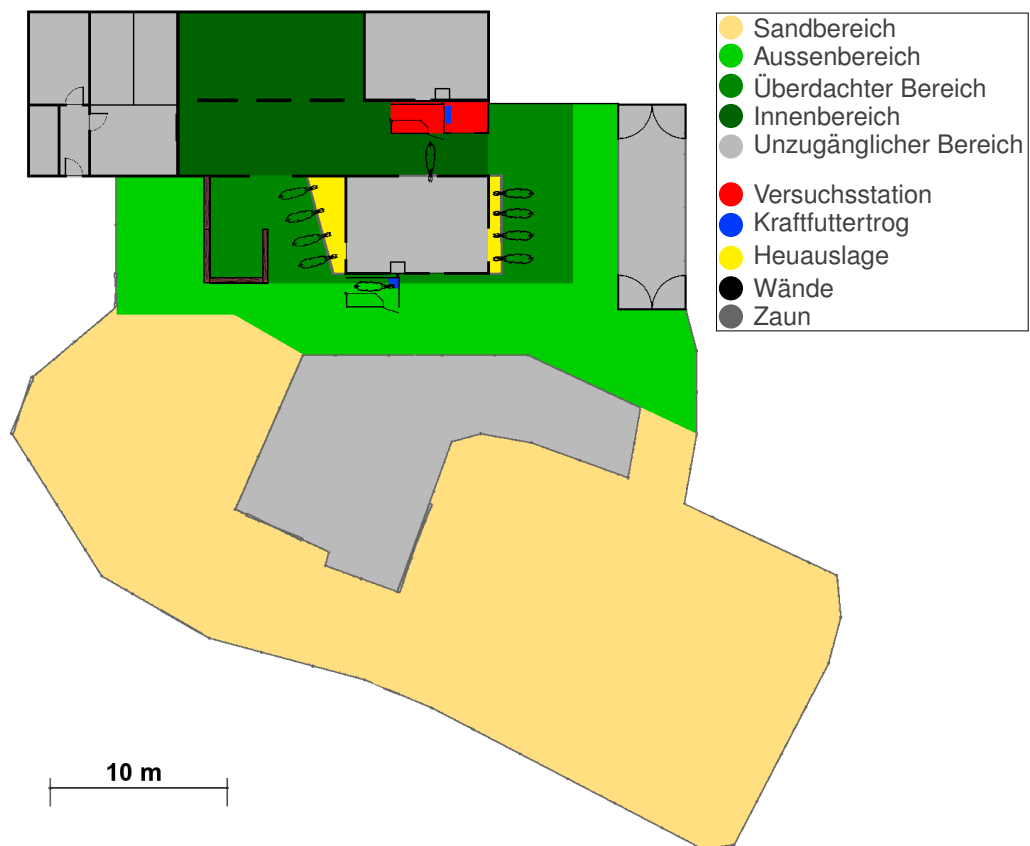


Abbildung 3.2.: Offenstall Grundriss [Verändert nach: Hinrichs Innovation + Technik GmbH 2012]

Die in der Abbildung 3.2 rot markierte Fläche beschreibt die Position der Versuchstation bzw. der mit dem Kameraerfassungssystem (Aufnahmeeinheit) ausgestatteten Abruffütterung im Stall. Die Station wurde im Inneren des Gebäudes aufgebaut, um einen möglichst geringen Einfluss durch einfallendes Sonnenlicht bei der Aufnahme zu gewährleisten. Direkt neben der Station befindet sich zwar eines der Tore, die ständig offen stehen, jedoch ist durch die Überdachung sichergestellt, dass direktes Sonnenlicht nur bei sehr tief stehender Sonne in den Aufnahmebereich einfallen kann. Die Wände der Abruffütterungsstation schützen zusätzlich vor störenden Lichteinflüssen innerhalb derselben.

Die Abruffütterungsstation (Abbildung 3.3a/b) wurde von der Firma HIT aufgebaut und mit einem Leitsystem versehen. Der vordere Bereich (B in Abb. 3.3) ist so konzipiert, dass das Tier mittels RFID-Technik (radio-frequency identification) erkannt wird, sobald es den Stand betritt. Der Bereich vor dem Futtertrog (c in Abb. 3.3) wird dann durch eine Nachlaufsperr (f in Abb. 3.3) abgeriegelt. Durch die Nachlaufsperr wird verhindert, dass das Tier, welches sich im Stand befindet, von einem ranghöheren Tier vertrieben wird. Das Tier kann so in Ruhe die abgerufene Futterration fressen. An der zunächst als „Standardversion“ zur Verfügung stehenden Abrufstation (siehe Abbildung 3.4b) wurden Änderungen vorgenommen, so dass der Trog (c in Abb. 3.3) sich nicht direkt über dem Boden, seitlich zur festen Wand hin, befindet, sondern in 40 cm Höhe über dem Boden mittig in der Station ausgerichtet ist. Zudem wurde ein V-Ausschnitt (d in Abb. 3.3) vor dem Trog aufgebaut, der durch eine Wand dem Tier den Zugang zum Trog verwehrt. Die Aufnahmeeinheit (b in Abb. 3.3), welche im Unterkapitel 3.3 näher beschrieben wird, ist direkt gegenüber dem V-Ausschnitt im Aufnahmebereich (A in Abb. 3.3) in einer Höhe von 1,29 m montiert. Hat das erkannte Pferd Anspruch auf Futter, fährt die Wand nach unten in den V-Ausschnitt und das Tier kann seinen Kopf durch den V-Ausschnitt stecken. Die Datenaufnahme beginnt, sobald ein Objekt hinter der herunterfahrenden Wand erkannt wird. Der V-Ausschnitt hat dabei für die Aufnahmen den Vorteil, dass der Bewegungsfreiraum des Kopfes ausreichend eingeschränkt ist. Die Tiere sind, zumindest vor oder nach der Futteraufnahme, gezwungen direkt in die Aufnahmeeinheit zu schauen. Hierdurch entstehen in kürzerer Zeit mehr frontale Aufnahmen der Pferdeköpfe, als dies in den kommerziell erhältlichen Standardstationen möglich wäre. Durch einen genügend großen Abstand des V-Ausschnitts zur Aufnahmeeinheit wird diese, sowie die Beleuchtung (a in Abb. 3.3), vor dem Zugang durch die Pferde geschützt. Der Bereich zwischen dem V-Ausschnitt und der Aufnahmeeinheit (A in Abb. 3.3) wird durch die Beleuchtung gut ausgeleuchtet und es kann durch Wände das Einfallen von störendem Licht verringert werden. Umgekehrt werden dadurch, dass die zusätzlich verbauten Geräte für die Tiere nicht zugänglich sind, Verletzungen verhindert.

3. Material und Methoden

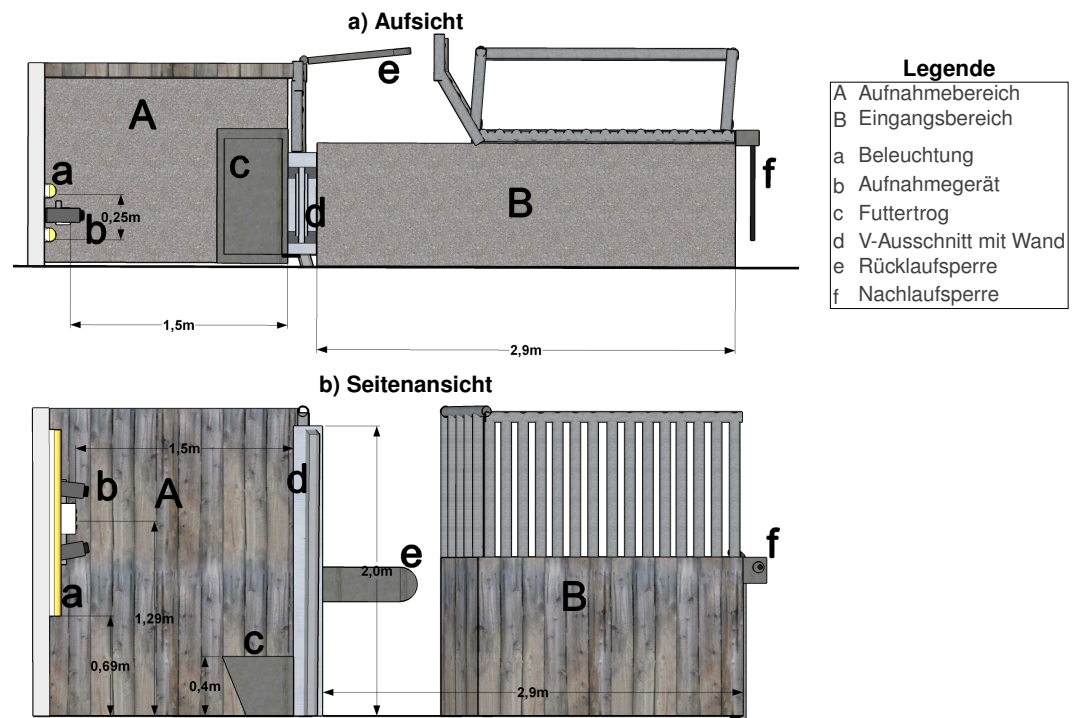


Abbildung 3.3.: Versuchsstation a) Aufsicht, b) Seitenansicht

Die Abbildung 3.4 zeigt auf der linken Seite (Abbildung 3.4a) ein Leitsystem, jedoch ohne den Futtertrog und den Aufnahmebereich, und auf der rechten Seite (Abbildung 3.4b) einen Futterstand in der bisher üblichen kommerziellen Standardausführung. Das Leitsystem aus Abbildung 3.4a ist mit dem Ausgang zur rechten Seite (genau entgegen der Abbildung) in der Versuchsstation verbaut.

In Abbildung 3.5a ist die Versuchsstation mit Tier von hinten oben fotografiert abgebildet. Abbildung 3.5b zeigt die Station mit Tier von oben aufgenommen.

3.2. Versuchsstall / Messanordnung

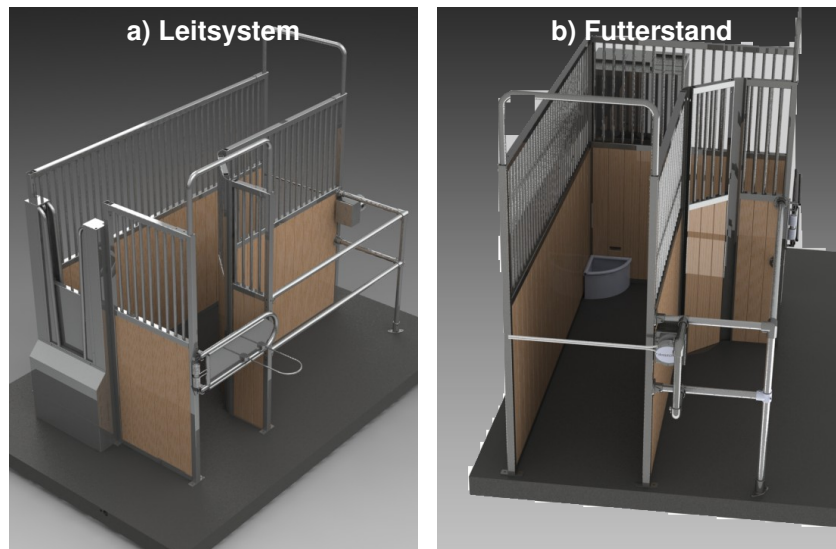


Abbildung 3.4.: Leitsystem a) mit V-Ausschnitt, b) bestehende Futterstation [Hinrichs Innovation + Technik GmbH 2012]



Abbildung 3.5.: Originalaufnahmen der Versuchsstation a) Rückansicht, b) Aufsicht

3.3. Bilderfassung

Zur Aufnahme der Pferdeköpfe wurde im Rahmen der vorliegenden Dissertation eine Aufnahmeeinheit entwickelt, welche in Abbildung 3.6 gezeigt ist. Die für den Aufbau der Einheit benötigten Komponenten sind in Tabelle 3.2 aufgeführt.

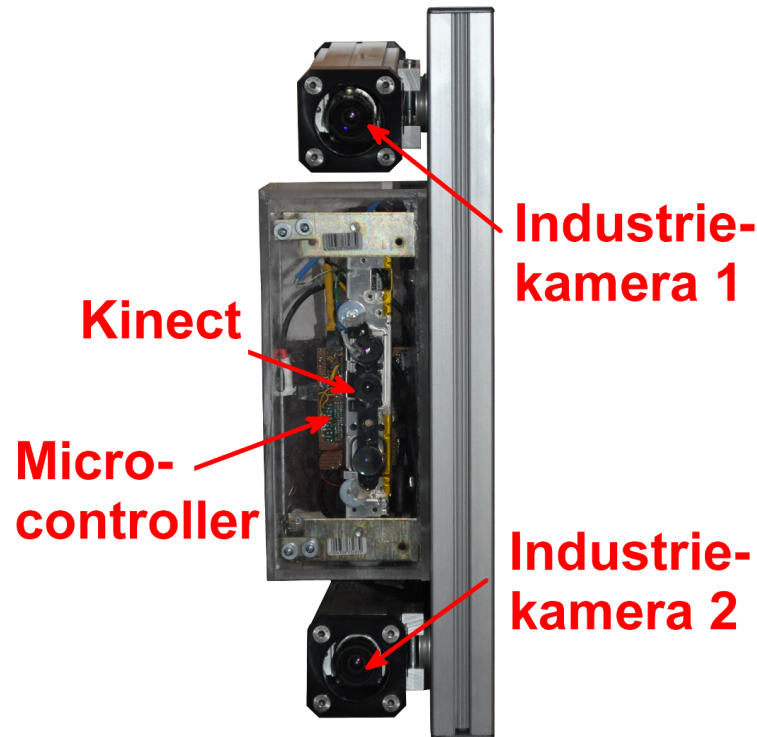


Abbildung 3.6.: Bild der Aufnahmeeinheit

Die Pferdeköpfe werden durch zwei die Aufnahmeeinheit einrahmenden Leuchtstoffröhren beleuchtet, da diese großflächig eine homogene Ausleuchtung des Aufnahmebereiches innerhalb der Station ermöglichen. Zudem ist der Aufnahmebereich durch Wände soweit von Außenlicht abgeschattet, dass die Lichtverhältnisse durch die künstliche Beleuchtung ausreichend konstant gehalten werden können (siehe Unterkapitel 3.2). Die Leuchtstoffröhren wurden hochkant mit einem Abstand von 25 cm voneinander an der Wand gegenüber dem V-Ausschnitt angebracht. Durch das Anbringen der Beleuchtung um die Aufnahmeeinheit herum kommt es nur bei extremen Positionen des Kopfes zur Schattenbildung auf den Pferdeköpfen. Die Leuchtstoffröhren werden mit einem EVG (elektronisches Vorschaltgerät) betrieben, welche die Frequenz der Netzspannung von 50 Hz auf etwa 40 kHz erhöht. Dies ist notwendig, da beim Betrieb ohne das EVG durch die niedrigere Frequenz bei der Aufnahme einer Reihe von Bildern durch die Industriekameras Helligkeitsschwankungen auftreten.

Tabelle 3.2.: Gerätespezifikation Aufnahmeeinheit

Beschreibung	Bezeichnung / Firma	Details
2 Leuchtstoffröhren	5LS 4127-1C / Siteco	Beleuchtung durch Leuchtstoffröhren mit EVG (elektronisches Vorschaltgerät, $\sim 100Hz$ auf Beleuchtungsintensität) Länge 120 cm , 36 W
Kinect	Kinect for Xbox / Microsoft	Nimmt Tiefenbild der Szene auf
2 Industriekameras	UI-6250SE-M-GL / IDS (Imaging Development Systems)	Monochromatische Kameras mit einer Auflösung von 1600×1200 und globalem Shutter um Bewegungsartefakte zu mindern
2 Objektive	DF6HA-1B / Fujinon	Kameraobjektive der Brennweite 6 mm (Winkel des Sichtbereiches bei $1/2'' : 56^\circ 09' \times 43^\circ 36'$)
2 UV-IR Sperrfilter	486 UV-IR Sperrfilter / B+W	Filter, die das Licht außerhalb der Wellenlängen $370\text{ nm} - 720\text{ nm}$ sperren
2 Schutzgehäuse	Salamander IP67 Schutzgehäuse / autoVimation	Schutzgehäuse für Industriekameras
Mikrocontroller	ATmega128L / Atmel	Steuert Aufnahme der Industriekameras in Abhängigkeit von der Aufnahme des Tiefenbildes
Rechner	NISE-3140 / OMTEC	Speichern und verarbeiten der Daten (Intel Core2Duo M P8400, 4 GB Arbeitsspeicher)

3. Material und Methoden

Die Aufnahmeeinheit besteht aus einer kommerziell erhältlichen Kinect für Xbox (Videospielkonsole von Microsoft) und 2 Industriekameras (Tabelle 3.2). Die Kinect liefert 30 Tiefenbilder der Szene pro Sekunde, die eine Auflösung von 640×480 Pixeln haben. Die Kinect projiziert dazu ein Muster (strukturiertes Licht) mit einer Wellenlänge von 830 nm (NIR-Licht, Nah-Infrarot-Licht, für das menschliche Auge nicht sichtbar) und nimmt dieses Muster mit einer IR-Kamera (Infrarot-Kamera) auf. Zur Bestimmung des Tiefenbildes wird die Disparität (Versatz) jedes Pixels des projizierten Musters zu dem Muster einer bekannten Entfernung bestimmt (dies geschieht bereits in der Kinect). Das Tiefenbild wird aus dem Disparitätenbild, welches eine Bittiefe von 11 Bit hat, errechnet. Zudem nimmt die Kinect Farbbilder mit der selben Frequenz und Auflösung wie die Tiefenbilder auf. Diese entsprechen durch den Transfer als verlustbehaftet komprimierte Bilder in der Qualität einem Webcam-Bild. Als Alternative lässt sich die Auflösung der Farbbilder verdoppeln (1280×1024), jedoch wird hierdurch die Aufnahme der Farb- und Tiefenbilder durch die beschränkte Bandbreite der Verbindung (USB) zum PC auf 15 FPS (Frames pro Sekunde) limitiert. Zur Aufnahme der Daten wurde die geringere Auflösung (640×480) mit einer höheren Framerate (30 FPS) genutzt.

Die verwendeten Industriekameras werden entsprechend der in Abschnitt 3.3.1 beschriebenen Methode durch den Mikrocontroller getriggert. Dies ist nötig um eine synchrone Datenaufnahme zu gewährleisten, da ein zeitlicher Versatz bei bewegten Objekten einen örtlichen Versatz beim Transfer der Tiefeninformation in die Bildebenen der Kameras zur Folge hat. Um einen entsprechenden Transfer der Tiefeninformation in eine der Kameraebenen berechnen zu können, sind Kameraparameter erforderlich, welche mittels der Kamerakalibrierung aus Unterkapitel 3.3.2 ermittelt wurden.

Da die Industriekameras, wenn auch nur geringfügig, sensibel im NIR-Bereich sind, wurden die Objektive mit UV-IR Sperrfiltern ausgestattet. Diese Sperren das Licht unterhalb von 370 nm und oberhalb von 720 nm und bilden einen ausschließlichen Bandpass für das sichtbare Licht.

Die im Stall vorherrschenden Bedingungen machen es notwendig, die Komponenten der Aufnahmeeinheit vor Staub, Feuchtigkeit, etc. und den Tieren zu schützen. Hierzu wurden die Kinect und der Mikrocontroller in einem Plexiglasgehäuse verbaut. Die Industriekameras inklusive der Objektive und Filter wurden jeweils in die vorgefertigten Schutzgehäuse eingefasst. Das Plexiglasgehäuse, sowie die beiden Schutzgehäuse, sind verstellbar auf einem Trägerprofil befestigt (siehe Abbildung 3.6).

3.3.1. Synchrone Datenaufnahme

Das synchrone Aufnehmen der Kamerabilder mit dem Tiefenbild ist notwendig, um Artefakte durch Bewegung des aufzunehmenden Objektes zu verhindern. Ein zeitlicher Versatz der Aufnahmen bedeutet beim Transformieren der Tiefeninformation in die Kamerabilder einen örtlichen Versatz. Hierdurch kommt es zu fehlerhaften

Segmentierungen, da sich das Objekt nicht an der durch die Tiefeninformation bestimmten Position im Bild befindet, sondern sich bereits von dieser Position entfernt hat.

Die synchrone Datenaufnahme wird zum einen durch ein Triggersignal (siehe $S_{K_{IR}}$ in Abbildung 3.7) auf der Hardwareebene sichergestellt. Zudem werden in einem Synchronisationszyklus zusammengehörende Frames optisch markiert (siehe $L_{K_{IR}}$ und L_{K_1} in Abbildung 3.8(a)). Die Markierungen werden benötigt, um die synchron aufgenommenen Frames auf der Softwareebene identifizieren zu können. Sie gewährleisten, dass, obwohl die Bilddaten durch unterschiedliche Transferraten mit zeitlichem Versatz im Rechner eintreffen, die Frames der unterschiedlichen Kameras einem Aufnahmezeitpunkt eindeutig und nachvollziehbar zugeordnet werden können.

Auf der Hardwareseite wird der Aufnahmezeitpunkt des Tiefenbildes von dem Mikrocontroller durch das Signal $S_{K_{IR}}$ an der IR-Kamera der Kinect bestimmt. Der IR-Kamera-Sensor MT9M001 [Widenhofer 2010; Chow u. a. 2012] und der Punkt, an dem das Signal $S_{K_{IR}}$ auf der Elektronik der Kinect abgenommen wird, sind in Abbildung 3.7 gezeigt. Das Signal $S_{K_{IR}}$ ist dabei verbunden mit dem *FRAME_VALID* Pin des Sensors MT9M001 aus dem Datenblatt [Micron 2006].

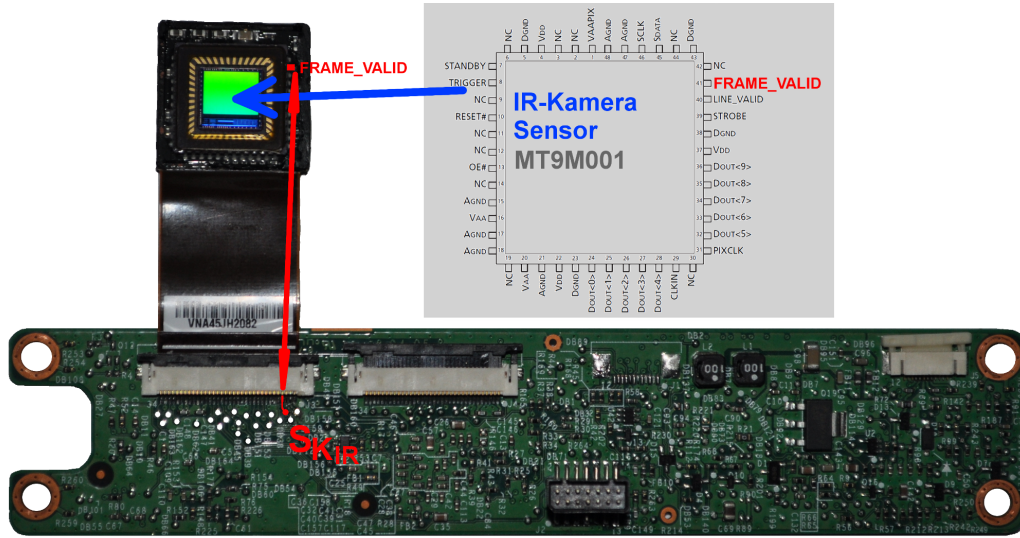


Abbildung 3.7.: Hardware-Triggersignal Abnahme auf der Kinect [Verändert nach Bildquelle: Micron 2006, Pinbelegung des Sensors, Seite. 5]

Der Mikrocontroller triggert, entsprechend der Schemata in Abbildung 3.8, die Aufnahme der beiden Industriekameras durch die Signale S_{K_1} und S_{K_2} , sobald der Aufnahmezeitpunkt des Tiefenbildes durch das Signal $S_{K_{IR}}$ an der Kinect detektiert wurde.

3. Material und Methoden

Da das Triggern der Aufnahmen der eingesetzten Industriekameras nicht mit einer Framerate von 30 FPS (Frames pro Sekunde) möglich ist, werden lediglich alle N Frames die Triggersignale S_{K_1} und S_{K_2} weitergeleitet. Die Anzahl N der zu überspringenden Frames kann über einen Befehlssatz auf dem Messwerterfassungs- und Steuerungsrechner an den Mikrocontroller übergeben werden (Einstellbar: $N \in (3, 4, \dots 10, 12)$).

Die maximale Framerate der Industriekameras beträgt 12.3 FPS [siehe Kameraspezifikationen: IDS 2013] bei einer Belichtungszeit von 1 ms im Triggermodus. Die Anzahl der möglichen Frames pro Sekunde (FPS) richtet sich bei den Kameras stark nach der Belichtungszeit. In allen durchgeführten Versuchen der vorliegenden Arbeit wurden 5 Bilder pro Sekunde (FPS) mit einer Belichtungszeit von 25 ms (erste Aufnahmen mit 100 ms und entsprechend angepasster Blende waren durch die Bewegung der Tiere deutlich verschwommen) aufgenommen, hierzu wurde der Frameteiler im Mikrocontroller auf $N = 6$ eingestellt, welches eine stabile Aufnahme (ohne eine größere Anzahl nicht verarbeiteter Triggersignale) der Bilder gewährleistete.

In dem Synchronisationszyklus werden ergänzend optische Markierungen auf dem IR-Kamerabild und dem Bild der ersten Industriekamera gesetzt. Die Markierungen, welche ebenfalls auf dem Tiefenbild sichtbar sind, sind in Abbildung 3.9 gezeigt. Sie werden durch LEDs erzeugt, welche durch die Signale $L_{K_{IR}}$ und L_{K_1} aus Schema 3.8(a) kurz aufblitzen. Die Leuchtdioden sind außen vor den Kameras angebracht, so dass sie nicht im Bild zu sehen sind, jedoch das Aufblitzen durch die Kamerasensoren wahrgenommen wird. Die LED vor der IR-Kamera der Kinect emittiert dabei Licht im IR-Bereich und die LED vor der Industriekamera im sichtbaren Wellenlängenbereich, um das emittierte Licht durch den jeweiligen Filter auf den Kamerasensoren erkennen zu können. Auf der Softwareebene werden nun die Frames mit entsprechenden LED-induzierten Markierungen als synchron aufgenommen registriert. Die Zuordnung der Frames der zweiten Industriekamera zu denen der ersten Kamera wird über die mit dem PC synchronisierte Systemzeit durchgeführt. Die Zuordnung der Farbbilder der Kinect geschieht lediglich über deren Ankunftszeit auf dem Messwerterfassungsrechner und wird mit der Ankunftszeit der Tiefenbilder verglichen.

Nachdem die Synchronität der einzelnen Kameradaten durch den beschriebenen Synchronisationszyklus aus Schema 3.8(a) einmal auf der Softwareseite registriert wurde, wird die synchrone Datenaufnahme durch das Schema 3.8(b) sichergestellt. Alle auf die Synchronisation folgenden Frames können dabei auf Softwareebene über ihre Framenummern eindeutig zugeordnet werden. Die Signale $L_{K_{IR}}$ bzw. L_{K_1} , die im Synchronisationszyklus die Marker setzten (siehe Schema 3.8(a)), werden nun ausgesetzt. Dabei wird der Zeitstempel der Kinect in eine entsprechende Framenummer umgerechnet. Selbst beim Verlust von einzelnen Frames, durch beispielsweise eine fehlerhafte Übertragung, bei der ein Frame nicht oder zu spät am Rechner ankommt, kann so eine Zuordnung der zeitgleich getriggerten Daten gewährleistet werden.

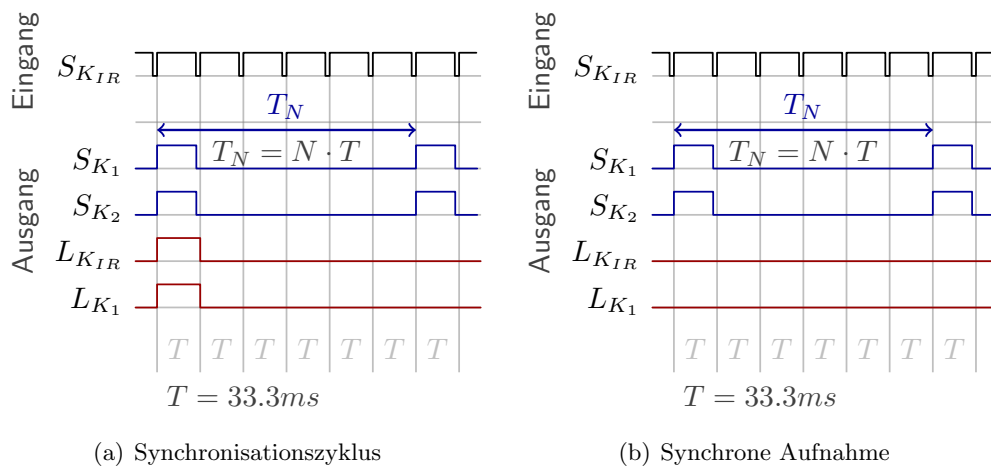


Abbildung 3.8.: Schema der Signale am Mikrocontroller

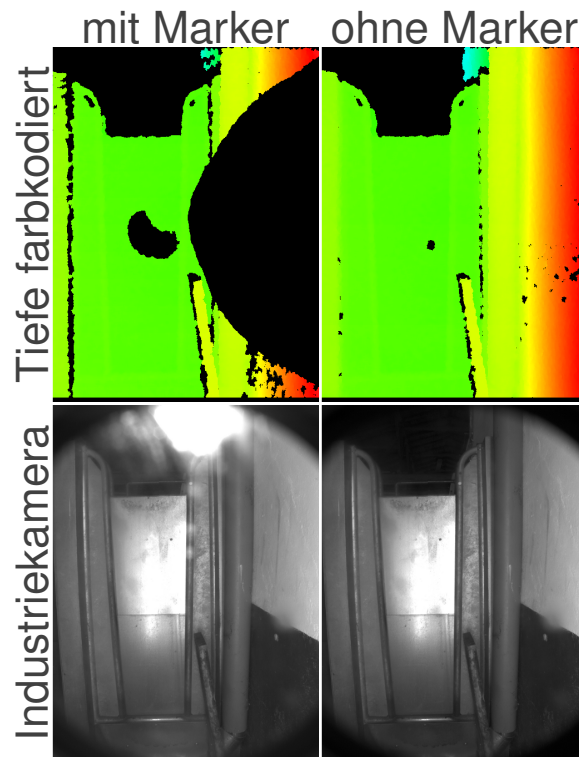


Abbildung 3.9.: Markierungen der LEDs auf den Kamerabildern

3.3.2. Kamerakalibrierung

Die Kameras der in diesem Kapitel beschriebenen Aufnahmeeinheit blicken aus unterschiedlichen Positionen im Raum auf die Szene bzw. den Pferdekopf. So haben keine zwei Kameras ein gleiches Bild der Szene. Dies schließt die IR-Kamera, aus der das Tiefenbild errechnet wird, ein. Eine Fusion der Daten, um beispielsweise ein über die Tiefe segmentiertes Objekt auf den Kamerabildern wiederzufinden, ist durch eine Transformation der Tiefeninformation auf die restlichen Kamerabilder möglich. Hierzu ist es jedoch notwendig, die Orientierung der Kameras im Raum und deren Eigenschaften zu kennen; dies erfolgt über die Kamerakalibrierung.

Die Kamerakalibrierung basiert auf dem im Abschnitt „*Kameramodell*“ beschriebenen Modell, welches die Eigenschaften der Abbildung beschreibt. Detailliertere Information zu der hier zusammengetragenen Kurzfassung finden sich in Stahl [2007] und Hartley u. Zisserman [2004]. Zudem treten durch den Einsatz von Objektiven nicht-lineare Verzerrungen bei der Abbildung auf, welche die linearen Modelle aus Abschnitt „*Kameramodell*“ erweitern und unter „*Verzerrung*“ beschrieben sind. Im letzten Unterabschnitt „*Kamerakalibrierung*“ werden Vorgehensweisen zur Bestimmung der Kameraparameter aufgezeigt.

Kameramodell

Projektive Kameras beschreiben die Abbildung $\mathcal{P}_{\mathcal{R}}$ eines Punktes $\mathbf{X} = [X \ Y \ Z]^T \in \mathbb{R}^3$ der Szene (3D Raum \mathbb{R}^3) auf einen Punkt $\mathbf{x}_{\mathcal{R}} = [u_{\mathcal{R}} \ v_{\mathcal{R}}]^T \in \mathbb{R}^2$ der Bildebene \mathcal{R} (Kameraebene \mathbb{R}^2) mit $\mathcal{P}_{\mathcal{R}} : (X, Y, Z) \in \mathbb{R}^3 \mapsto \mathcal{P}_{\mathcal{R}}(X, Y, Z) := (u_{\mathcal{R}}, v_{\mathcal{R}}) \in \mathbb{R}^2$. Die projektiven Kameras teilen sich in zwei Hauptklassen, die affinen und die perspektivischen Kameras.

Affine Kameras führen eine parallele Projektion durch, welche Parallelitäten erhält. Bei der Verwendung eines beidseitig telezentrischen Objektivs beispielsweise verlaufen die Hauptstrahlen parallel zur optischen Achse. In den Modellen der affinen Kameras befindet sich das Kamerazentrum im Unendlichen, so hat die Entfernung eines Objektes bei der Abbildung keinen Einfluss auf dessen Abbildungsgröße; im Gegensatz zu den perspektivischen Kameras, die in der vorliegenden Arbeit verwendet wurden. Diese führen eine zentrale Projektion durch und haben ihr Kamerazentrum im Endlichen. Sie werden mit dem prinzipiellen Modell der Lochkamera beschrieben. Bei dieser Art Projektion verhält sich die Entfernung eines Objektes zur Kamera antiproportional zu dessen Abbildungsgröße auf der Kameraebene.

Eine „einfache“, durch das Lochkameramodell beschriebene, perspektivische Projektion ist in Abbildung 3.10 gezeigt. In der Abbildung sind das Kamerazentrum \mathbf{C} , welches dem perspektivischen Projektionszentrum entspricht, und die Bildebene \mathcal{R} , auch retinale Ebene genannt, des Lochkameramodells zu erkennen. Der Schnittpunkt der Verbindungslinie zwischen dem Punkt \mathbf{X}_{cam} des Kamerakoordinatensystems und dem Kamerazentrum \mathbf{C} mit der retinalen Ebene \mathcal{R} entspricht dabei dem Punkt $\mathbf{x}_{\mathcal{R}}$,

der Projektion von \mathbf{X}_{cam} . Die Optische Achse verläuft senkrecht zur retinalen Ebene und schneidet das Kamerazentrum. Auf ihr findet sich die Strecke, welche den kleinsten Abstand f zwischen der Ebene \mathcal{R} und dem Punkt \mathbf{C} beschreibt. f wird auch als Brennweite bezeichnet, wobei dieses lediglich für Linsensystem korrekt ist.

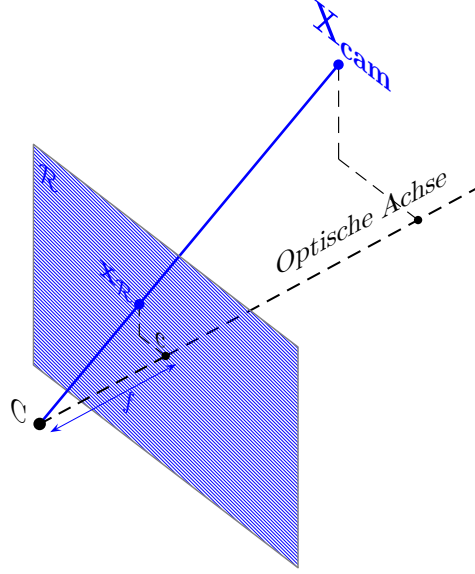


Abbildung 3.10.: Perspektivische Projektion [Verändert nach: Stahl 2007]

Für den Fall, dass sich das Kamerazentrum im Ursprung des Welt-Koordinatensystems befindet ($\mathbf{C} = [0 \ 0 \ 0]^T$), gilt $\mathbf{X}_{\text{cam}} = \mathbf{X}$. Die Projektion von $\mathbf{X}_{\text{cam}} = [X_{\text{cam}} \ Y_{\text{cam}} \ Z_{\text{cam}}]^T$ auf $\mathbf{x}_{\mathcal{R}} = [u_{\mathcal{R}} \ v_{\mathcal{R}}]^T$ lässt sich dann durch die Formel

$$u_{\mathcal{R}} = f \frac{X_{\text{cam}}}{Z_{\text{cam}}}, \quad v_{\mathcal{R}} = f \frac{Y_{\text{cam}}}{Z_{\text{cam}}} \quad (3.1)$$

beschreiben. Mit Hilfe der projektiven Geometrie, welche im Anhang B kurz erläutert ist, kann die Projektion vereinfacht durch eine Matrixmultiplikation dargestellt werden:

$$\begin{pmatrix} u_{\mathcal{R}} \\ v_{\mathcal{R}} \\ 1 \end{pmatrix} \sim \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X_{\text{cam}} \\ Y_{\text{cam}} \\ Z_{\text{cam}} \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \mathbf{P}_0 \check{\mathbf{X}}_{\text{cam}} = \mathbf{K} \mathbf{P}_0 \check{\mathbf{X}}_{\text{cam}} \quad (3.2)$$

Die Koordinatenpunkte im projektiven Raum sind mit $\check{}$ gekennzeichnet.

3. Material und Methoden

Die gesamte Projektionsmatrix teilt sich in die eigentliche Projektionsmatrix

$$\mathbf{P}_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (3.3)$$

und die interne Kameramatrix

$$\mathbf{K} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (3.4)$$

Die intrinsischen Parameter, welche die Eigenschaften der Kamera beschreiben, werden zum Aufbau der internen Kameramatrix verwendet. Die bisherige Projektion $\mathbf{x}_{\mathcal{R}}$ befindet sich in einem metrischen Koordinatensystem. Um eine vollständige Abbildung eines Punktes auf die jeweiligen Pixel des Kamerasensors zu beschreiben, ist eine Erweiterung der in Gleichung 3.2 verwendeten Kameramatrix nötig. Hierzu werden die Ausdehnungen der Sensor-Pixel berücksichtigt, so dass sich die Projektion wie folgt verändert:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \sim \mathbf{K} \mathbf{P}_0 \begin{pmatrix} X_{cam} \\ Y_{cam} \\ Z_{cam} \\ 1 \end{pmatrix}, \quad \text{mit} \quad \mathbf{K} = \begin{pmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (3.5)$$

u und v ($u, v \in \mathbb{N}_0$) sind die jeweiligen Pixel, auf die der Punkt \mathbf{X}_{cam} mit $\mathcal{P}_I : (X_{cam}, Y_{cam}, Z_{cam}) \in \mathbb{R}^3 \mapsto \mathcal{P}_I(X_{cam}, Y_{cam}, Z_{cam}) := (u, v) \in \mathbb{N}_0^2$ abgebildet wird. α_x berechnet sich dabei durch $\frac{f}{p_x}$, mit p_x der Pixelgröße des Kamerasensors in x-Richtung. Entsprechend gilt $\alpha_y = \frac{f}{p_y}$, mit p_y für die y-Richtung. s ist ein zusätzlicher Faktor, der eine Scherung durch eine nicht rechtwinklige Ausrichtung der Pixel auf dem Sensor korrigiert. Zumeist sind die Kamerapixel jedoch rechtwinklig zueinander ausgerichtet, wodurch s nahe Null ist. Durch die Verschiebung um x_0 und y_0 wird der bisherige Nullpunkt auf der Ebene vom Hauptpunkt (dem Schnittpunkt der Kameraebene mit der optischen Achse) in die obere linke Ecke verschoben. Eine detailliertere Beschreibung findet sich, wie bereits erwähnt, in [Stahl 2007].

Da sich das Kamerazentrum \mathbf{C} im Allgemeinen nicht im Ursprung des Weltkoordinatensystems befindet, ist es zuvor notwendig, einen Punkt \mathbf{X} der Szene in das Kamerakoordinatensystem \mathbf{X}_{cam} zu transformieren. Dies entspricht der Abbildung $\mathcal{P}_E : (X, Y, Z) \in \mathbb{R}^3 \mapsto \mathcal{P}_E(X, Y, Z) := (X_{cam}, Y_{cam}, Z_{cam}) \in \mathbb{R}^3$. Die Abbildung setzt sich zusammen aus einer Rotation \mathbf{R} und einer Transformation \mathbf{t} nach der Gleichung

$$\mathbf{X}_{cam} = \mathbf{R}\mathbf{X} + \mathbf{t}, \quad \text{mit} \quad \mathbf{t} = -\mathbf{R}\mathbf{C}. \quad (3.6)$$

Diese Transformation beschreibt die Orientierung der Kamera im Raum. Die entsprechenden Parameter werden als extrinsische Kameraparameter bezeichnet. \mathbf{R} ist

eine Rotationsmatrix, welche die Drehung des Kamerasystems relativ zum Weltkoordinatensystem angibt. Und \mathbf{C} ist das Kamerazentrum im Weltkoordinatensystem. Die Transformation vom Weltkoordinatensystem in das Kamerakoordinatensystem ist in Abbildung 3.11 gezeigt.

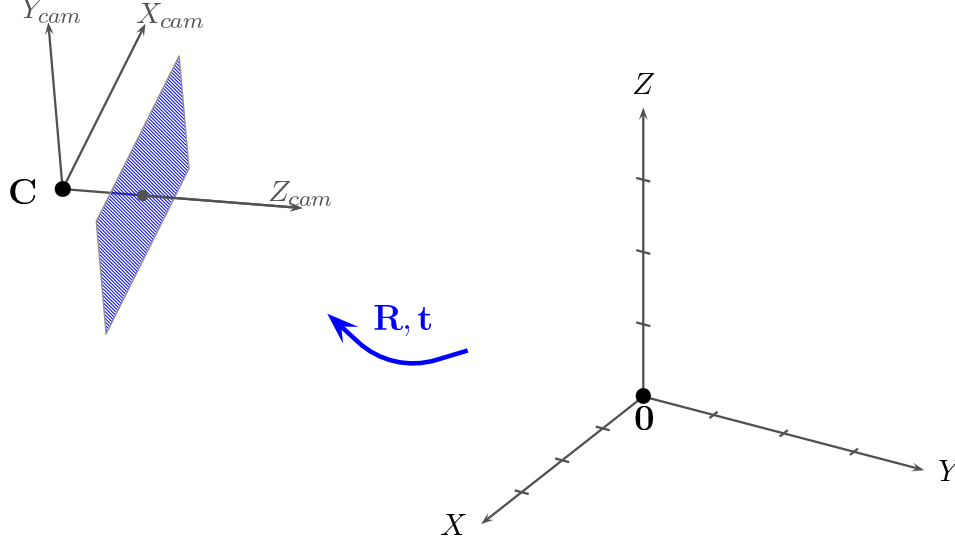


Abbildung 3.11.: Euklidische Transformation vom Welt- ins Kamerakoordinatensystem [Verändert nach: Stahl 2007]

In der projektiven Geometrie (siehe Anhang B) lässt sich die Transformation mit der Matrix

$$\check{\mathbf{X}}_{\text{cam}} \sim \begin{bmatrix} \mathbf{R} & -\mathbf{RC} \\ 0 & 1 \end{bmatrix} \check{\mathbf{X}} \quad (3.7)$$

beschreiben.

Die Kombination aus Gleichung 3.5 und Gleichung 3.7 ergibt die allgemeine Projektion:

$$\check{\mathbf{x}}_{\text{cam}} = \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \sim \begin{pmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{bmatrix} \mathbf{R} & -\mathbf{RC} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (3.8)$$

Sie entspricht der Komposition $\mathcal{P} : \mathcal{P}_I \circ \mathcal{P}_E$ wodurch sich die gesamte Abbildung \mathcal{P} zu $\mathcal{P} : (X, Y, Z) \in \mathbb{R}^3 \mapsto \mathcal{P}(X, Y, Z) := (u, v) \in \mathbb{N}_0^2$ ergibt.

Verzerrung

Die im Abschnitt „Kameramodell“ beschriebenen Abbildungen sind (idealisiert) linear. Durch den Einsatz von Objektiven treten jedoch häufig Verzerrungen auf, die nicht-linear sind. Diese Abbildungsfehler sind abhängig vom Abstand zur optischen Achse und können auf der Abbildungsebene \mathcal{R} korrigiert werden. Unterschieden wird vorrangig zwischen der radialen und der tangentialen Verzerrung, wobei die Korrektur der tangentialen Verzerrung auch zu numerischer Instabilität führen kann [Zhang 1999].

Die radiale Verzerrung wird nach Herrera u. a. [2011] durch

$$\tilde{\mathbf{x}}_{\text{rd}} = (1 + \kappa_1 r^2 + \kappa_2 r^4) \mathbf{x}_{\text{n}}, \quad \text{mit } r = \sqrt{u_n^2 + v_n^2}, \quad \mathbf{x}_{\text{n}} = \begin{bmatrix} u_n \\ v_n \end{bmatrix} \quad (3.9)$$

berechnet. $\kappa_{1,2}$ geben die Form und Stärke der Verzerrung an, r ist der Abstand zum Hauptpunkt (dem Schnittpunkt von optischer Achse und der Ebene \mathcal{R}). $\tilde{\mathbf{x}}_{\text{rd}} = [\tilde{u}_{\text{rd}} \ \tilde{v}_{\text{rd}}]^T$ beschreibt die tangential verzerrten Koordinaten von $\mathbf{x}_{\text{n}} = [u_n \ v_n]^T$. $\mathbf{x}_{\text{n}} = [u_n \ v_n]^T$ sind die normalisierten Koordinaten nach Gleichung 3.10, direkt nach der Projektion von \mathbf{X} auf die Bildebene \mathcal{R} , welche durch die Anwendung der extrinsischen Kameraparameter und der Projektion \mathbf{P}_0 berechnet wurden. Die intrinsischen Kameraparameter werden jedoch erst nach der Verzerrung auf die Koordinaten angewendet, da die Nullposition sich ansonsten nicht mehr auf den Hauptpunkt bezieht.

$$\mathbf{x}_{\text{n}} = \mathbf{P}_0 (\mathbf{R}\mathbf{X} + \mathbf{t}) \quad (3.10)$$

Sind die intrinsischen Parameter bereits angewendet, lässt sich \mathbf{x}_{n} aus $\mathbf{x} = [u \ v]^T$ berechnen:

$$\mathbf{x}_{\text{n}} = \begin{bmatrix} u_n \\ v_n \end{bmatrix} = \begin{bmatrix} \frac{u-x_0}{\alpha_x} \\ \frac{v-y_0}{\alpha_y} \end{bmatrix} \quad (3.11)$$

In Abbildung 3.12 sind unterschiedliche Formen der radialen Verzerrung dargestellt. Der Hauptpunkt, von dessen Abstand r der Fehler abhängt, befindet sich jeweils mittig im Bild. Für die gezeigten Fälle ist in der oberen Reihe $\kappa_2 = 0$, es sind somit nur die quadratischen Terme gezeigt. In der unteren Reihe ist $\kappa_1 = 0$, so dass der quartische Fehlerterm sichtbar ist. Auf der linken Seite findet sich eine kissenförmige Verzeichnung mit $\kappa_1 > 0$ (bzw. $\kappa_2 > 0$ für die zweite Reihe), in der Mitte ist das unverzerrte Bild $\kappa_1 = \kappa_2 = 0$ zu sehen und auf der rechten Seite ist eine tonnenförmige Verzerrung mit $\kappa_1 < 0$ (bzw. $\kappa_2 < 0$) gezeigt.

Die tangentiale Verzerrung wird nach Herrera u. a. [2011] durch

$$\tilde{\mathbf{x}}_{\text{td}} = \begin{bmatrix} 2\kappa_3 u_n v_n + \kappa_4 (r^2 + 2u_n^2) \\ \kappa_3 (r^2 + 2v_n^2) + 2\kappa_4 u_n v_n \end{bmatrix}, \quad \text{mit } r = \sqrt{u_n^2 + v_n^2}, \quad \mathbf{x}_{\text{n}} = \begin{bmatrix} u_n \\ v_n \end{bmatrix} \quad (3.12)$$

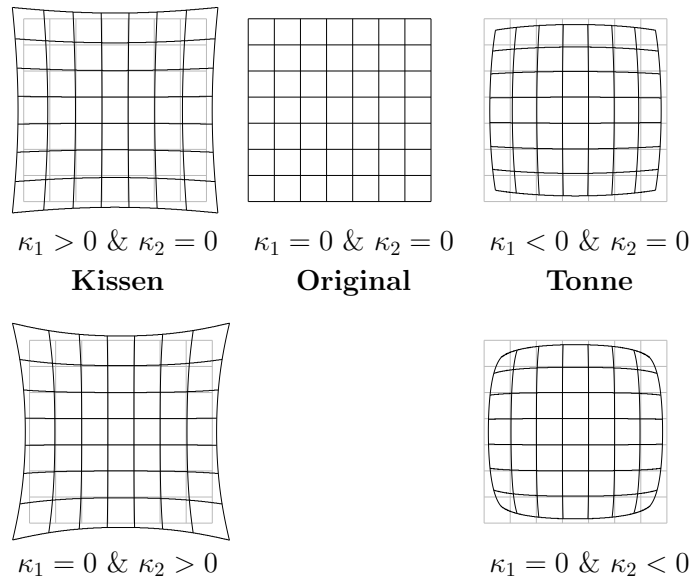


Abbildung 3.12.: Radiale Verzerrung

berechnet. Wie zuvor für die radiale Verzerrung beschrieben, sind $\mathbf{x}_n = [u_n \ v_n]^T$ die normalisierten Koordinaten und r der Abstand zum Hauptpunkt. In Abbildung 3.13 sind die einzelnen Fälle für κ_3 und κ_4 dargestellt. Der Hauptpunkt befindet sich wiederum mittig im Bild. Die tangentielle Verzerrung bietet die Möglichkeit Fehler, welche senkrecht zum radialen Fehler existieren, zu modellieren.

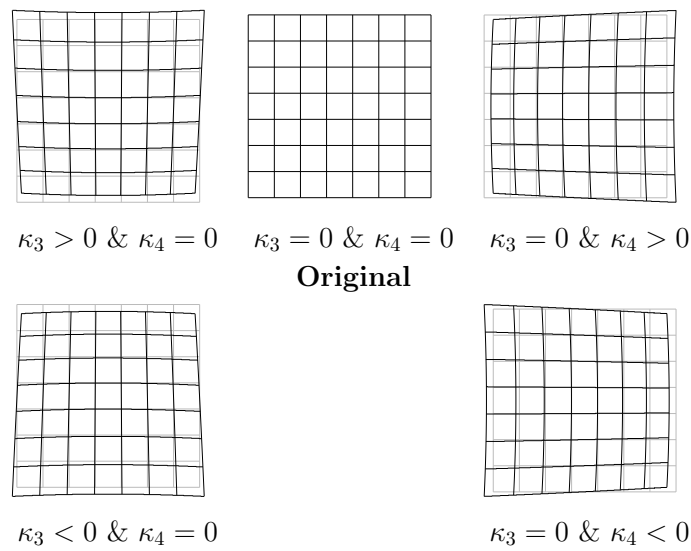


Abbildung 3.13.: Tangentielle Verzerrung

3. Material und Methoden

Die verzerrten Koordinaten $\tilde{\mathbf{x}}_{\mathbf{d}}$ werden aus der Kombination der radialen und tangentialen Verzerrung durch die Summe

$$\tilde{\mathbf{x}}_{\mathbf{d}} = \tilde{\mathbf{x}}_{\mathbf{rd}} + \tilde{\mathbf{x}}_{\mathbf{td}} \quad (3.13)$$

berechnet.

Kamerakalibrierung

Die in dieser Arbeit verwendete Kamerakalibrierung stammt von Herrera u. a. [2011], welche auf der Kalibrierung von Zhang [2004] aufbaut und diese für das Kalibrieren von Tiefenkameras erweitert. In [Herrera u. a. 2011] wird jeweils eine Grauwertkamera mit einer Tiefenkamera kalibriert. In dieser Arbeit wurde die von Herrera u. a. [2011] veröffentlichte Kalibrieroutine auf jede der Kameras angewendet. Um alle Kameras aus Kapitel 3.3 zu kalibrieren, wurden die Farb- und die zwei Industriekameras der Reihe nach einzeln mit der Tiefenkamera kalibriert. Die Kamerakalibrierung kann mit einfachen Kalibrierobjekten (z. B. einem ausgedruckten Schachbrett 2D) für einzelne Kameras durchgeführt werden [Zhang 1999, 2004]. In Zhang [2004] ist eine Übersicht der möglichen Kalibrierobjekte von 3D bis 1D Objekten und die Selbstkalibrierung mit dem entsprechenden Vorgehen zu finden. Zur Kalibrierung werden Punktkorrespondenzen zwischen den Punkten des Kalibrierobjektes \mathbf{X} im \mathcal{R}^3 und den durch die zu kalibrierende Kamera abgebildeten Punkten \mathbf{x} im \mathcal{R}^2 verwendet. Zhang [1999] bestimmt initial die intrinsischen Parameter α_x , α_y , s , x_0 und y_0 , sowie die sechs extrinsischen Parameter, mit Hilfe eines linearen Ausgleichsproblems. Hierzu wird zuerst über den Algorithmus „Direkte Lineare Transformation“ (DLT - direct linear transformation [Hartley u. Zisserman 2004]) eine Homografie (Bildet Punkte aus einem Raum in den selben ab) aus den Korrespondenzpunkten bestimmt. Aus dieser wird ein Ausgleichsproblem aufgestellt, welches durch Nebenbedingungen sicherstellt, dass die Zeilen der Rotationsmatrix orthonormal zueinander sind. Die durch die Initialisierungsmethode gewonnen Parameter sind jedoch nicht physikalisch anwendbar und stellen lediglich eine algebraisch minimierte Lösung dar. Zur genaueren Bestimmung der Parameter wird eine nichtlineare Fehlerfunktion wie in Gleichung 3.14 aufgestellt, die die absolute Differenz zwischen der Projektion der einzelnen Punkte des Kalibrierobjektes $\hat{\mathbf{x}}$ und deren Korrespondenzpunkten im Bild \mathbf{x} aufsummiert.

$$\sum_{n=1}^N \sum_{m=1}^M \|\mathbf{x}_{nm} - \hat{\mathbf{x}}(\alpha_x, \alpha_y, s, x_0, y_0, \kappa_1, \kappa_2, \mathbf{R}_n, \mathbf{t}_n, \mathbf{X}_m)\| \quad (3.14)$$

Bei der Projektion werden durch Zhang [1999] die Parameter der radialen Verzerrung κ_1 und κ_2 berücksichtigt, die der tangentialen Verzerrung κ_3 und κ_4 jedoch nicht. N ist die Anzahl der Bildaufnahmen und M die Anzahl der Korrespondenzpunkte. Die Fehlerfunktion wird dann durch ein nichtlineares Optimierungsverfahren (Levenberg-Marquardt Algorithmus [Hartley u. Zisserman 2004]) minimiert. Bei N

Aufnahmen sind $6N + 7$ Parameter zu bestimmen, sechs für die extrinsischen Parameter, welche sich in jedem Bild ändern, und sieben für die intrinsischen Parameter, welche zwischen den Aufnahmen konstant bleiben. Da für jede Punktkorrespondenz sich zwei Gleichungen ergeben, können $2MN$ Unbekannte bestimmt werden. Zur Kalibrierung aller Parameter reichen daher zwei Aufnahmen, auf denen mindestens fünf Punktkorrespondenzen extrahiert werden können, aus. Es ist jedoch darauf zu achten, dass die Punkte in beiden Aufnahmen nicht koplanar (einfache Translation) und die Ebenen des Kalibrierobjektes nicht parallel sind. Mit der zu bestimmenden Anzahl der intrinsischen Parameter ändert sich auch die minimal benötigte Anzahl an Punktkorrespondenzen M . Um eine ausreichende Genauigkeit zu erreichen, wird in Zhang [2004] empfohlen $N \geq 4 - 5$ Aufnahmen zur Kalibrierung zu verwenden. Zur Kalibrierung der Kameras aus Kapitel 3.3 wurden in dieser Arbeit jeweils 44 Aufnahmen pro Kamera verwendet. Zudem können mehrere Kameras kalibriert werden. Die Kameras befinden sich dabei, wie in Abbildung 3.14 dargestellt, in einer festen Position zueinander. Die Kameras können hierzu an einem Gestell befestigt sein. Dabei sollen jeweils die intrinsischen Parameter der Kameras bestimmt werden und ihre relative Anordnung zueinander \mathbf{R}^r und \mathbf{t}^r . Die Parameter der zweiten Kamera sind mit einem Hochkomma ' gekennzeichnet. Wie in Zhang [2004], werden meist alle Kameras einzeln kalibriert, um dann eine zu optimierende Fehlerfunktion für zwei Kameras (Stereo-System) aufzustellen:

$$\sum_{n=1}^N \sum_{m=1}^M \partial_{mn} \|\mathbf{x}_{nm} - \hat{\mathbf{x}}(\alpha_x, \alpha_y, s, x_0, y_0, \kappa_1, \kappa_2, \mathbf{R}_n, \mathbf{t}_n, \mathbf{X}_m)\|^2 + \partial'_{mn} \|\mathbf{x}'_{nm} - \hat{\mathbf{x}}'(\alpha'_x, \alpha'_y, s', x'_0, y'_0, \kappa'_1, \kappa'_2, \mathbf{R}'_n, \mathbf{t}'_n, \mathbf{X}_m)\|^2 \quad (3.15)$$

∂_{mn} und ∂'_{mn} haben den Wert 1 für den Fall, dass die entsprechende Punktkorrespondenz in der jeweiligen Aufnahme zu sehen ist, und andernfalls den Wert 0. Dadurch können Punktkorrespondenzen berücksichtigt werden, die nur in einem der Bilder zu sehen ist.

Durch die feste Position der Kameras zueinander, variieren \mathbf{R}^r und \mathbf{t}^r zwischen den Aufnahmen nicht. Dadurch kann \mathbf{R}' , entsprechend der Gleichung 3.16, durch die Kombination aus \mathbf{R} und \mathbf{R}^r beschrieben werden. Dies gilt ebenfalls für die Kombination von \mathbf{t} und \mathbf{t}^r .

$$\mathbf{R}' = \mathbf{R} \mathbf{R}^r \quad \text{und} \quad \mathbf{t}' = \mathbf{R} \mathbf{t}^r + \mathbf{t} \quad (3.16)$$

Daher sind bei der Kalibrierung statt $2(6N)$, nur $6N + 6$ unbekannte extrinsische Parameter zu bestimmen (N ist die Anzahl der Kalibrieraufnahmen einer Kamera). Hierzu würde wiederum für jede der Kameras die entsprechende Anzahl an intrinsischen Parametern kommen.

Die Kalibrierung einer Tiefenkamera mit einer Grau- oder Farbkamera stellt eine Besonderheit der Kalibrierung zweier Kameras dar. Abhängig von der Art der

3. Material und Methoden

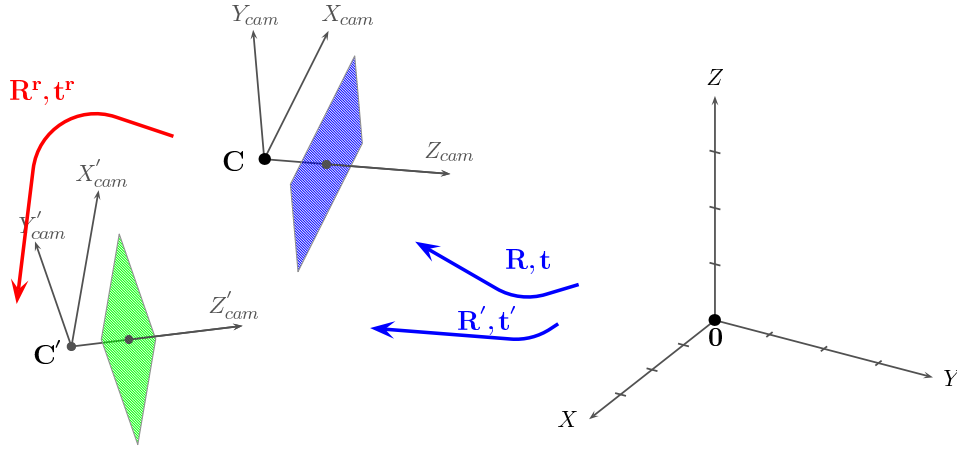


Abbildung 3.14.: Transformation zwischen Kamerakoordinatensystemen

Tiefenkamera, kommt es vor, dass die Punktkorrespondenzen des Kalibrierobjektes nicht im Tiefenbild zu erkennen sind. Dies ist in Abbildung 3.15a erkennbar, rechts davon (3.15b) ist das entsprechende Kalibrierobjekt aus Sicht der Grauwertkamera abgebildet. In solchen Fällen haben die zu kalibrierenden Kameras, wie in Abbildung 3.16 gezeigt, keine übereinstimmenden Bezugssysteme (Weltkoordinatensystem). Das Bezugssystem der Tiefenkamera ist dort in Grün zu erkennen, wohingegen das der Grauwertkamera in Blau mit dem angedeuteten Schachbrett dargestellt ist. Die in dieser Arbeit verwendete Kalibrierung ist in Herrera u. a. [2011] beschrieben und löst, wie im Folgenden erläutert, diese Problematik.

Die im Kapitel 3.3 beschriebene Tiefenkamera Kinect bestimmt durch ein Projektionsverfahren ein Disparitätenbild. Die Disparität d ist der Pixelversatz. Das Disparitätenbild hält in jedem Pixel in einer Art 2D-Array die entsprechende Disparität. Das Disparitätenbild beschreibt somit eine Abbildung $\mathcal{I}_D : (u, v) \in \mathbb{N}_0^2 \mapsto \mathcal{I}_D(u, v) := d \in \mathbb{Z}$ der Pixel auf den entsprechenden Pixelversatz. Die Disparität verhält sich im Allgemeinen antiproportional zur Entfernung des in dem jeweiligen Pixel aufgenommenen Objektes und der Kamera. Dadurch kann aus dem Disparitätenbild, mit entsprechender Kalibrierung, für jeden Pixel ein Punkt $\mathbf{X}_{\text{cam}}^d = [X_{\text{cam}}^d \ Y_{\text{cam}}^d \ Z_{\text{cam}}^d]^T$ im Tiefenkamerasystem errechnet werden. Ausnahmen sind hierbei die durch Verdeckung oder zu geringe Entfernung ungültigen Pixel (schwarze Bereiche in Abbildung 3.15a). Der Tiefenwert Z_{cam}^d wird dazu mit dem Disparitätswert d durch

$$Z_{\text{cam}}^d = \frac{1}{\beta(d - \gamma)} \quad (3.17)$$

modelliert. β und γ sind zusätzliche intrinsische Parameter der Tiefenkamera. Im Fall der Kinect verhält sich der Disparitätenwert d proportional zur Entfernung. Dies liegt an dem durch Gleichung 3.17 beschriebenen Modell und der dazugehörigen

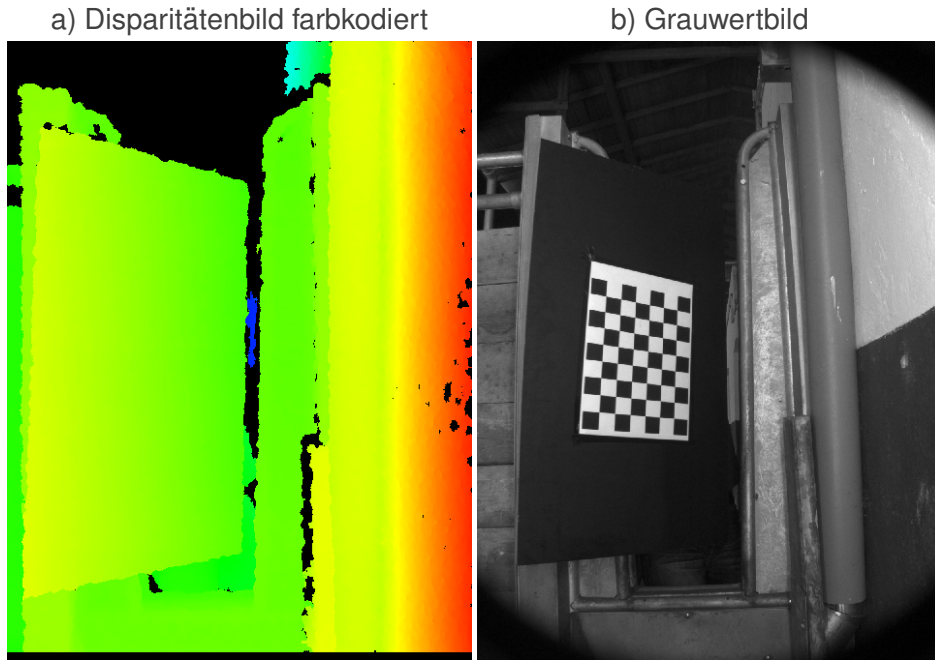


Abbildung 3.15.: Kamerabilder des Kalibrierobjektes [a) Tiefenkamera, b) Industriekamera]

intrinsischen Parameter ($\beta < 0$ und $d < \gamma$ für gültige Pixel). Die Transformation der Disparität auf die Tiefe entspricht der Abbildung $\mathcal{F}_Z : d \in \mathbb{Z} \mapsto \mathcal{F}_Z(d) := Z_{cam}^d \in \mathbb{R}$. Werden die Tiefenwerte wiederum in einem Bild (2D-Array) dargestellt, ergibt sich das Tiefenbild \mathcal{I}_Z durch die Komposition von $\mathcal{I}_Z : \mathcal{F}_Z \circ \mathcal{I}_D$. Dies bildet mit $\mathcal{I}_Z : (u, v) \in \mathbb{N}_0^2 \mapsto \mathcal{I}_Z(u, v) := Z_{cam}^d \in \mathbb{R}$ die jeweiligen Pixel auf die Tiefenwerte Z_{cam}^d ab.

Die Transformation eines Punktes $\mathbf{X}_{cam}^d = [X_{cam}^d \ Y_{cam}^d \ Z_{cam}^d]^T$ im Tiefenkamerakoordinatensystem auf den Pixel $\mathbf{x}^d = [u^d \ v^d]^T$ erfolgt wie beim Lochkamera-modell, welches im Unterkapitel „Kameramodell“ beschrieben ist.

3. Material und Methoden

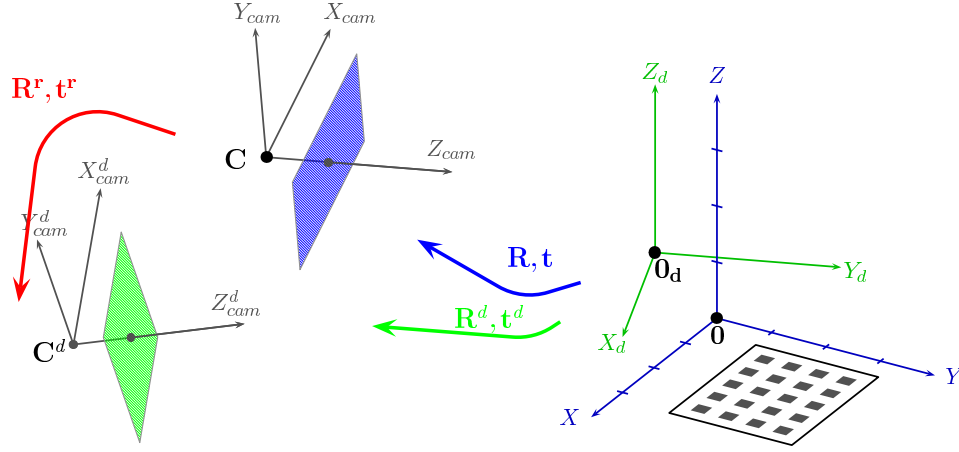


Abbildung 3.16.: Kamerakalibrierung mit unterschiedlichen Bezugssystemen

In einem ersten Schritt werden nach Herrera u. a. [2011] die jeweiligen Kameras (Tiefenkamera und Grauwertkamera) einzeln kalibriert. Die Kalibrierung der Grauwertkamera erfolgt, wie zuvor beschrieben, nach der linearen Initialisierungsmethode von Zhang [1999]. Es lassen sich dadurch die intrinsischen Parameter α_x , α_y , x_0 , y_0 und die extrinsischen Parameter \mathbf{R} und \mathbf{t} der Grauwertkamera bestimmen. Eine Scherung wird nicht angenommen ($s = 0$). Die Parameter der Verzerrung werden vorerst nicht berücksichtigt. Für eine grobe Kalibrierung der Tiefenkamera werden vom Benutzer auf dem Disparitätenbild die vier Eckpunkte der Ebene, auf der sich das Schachbrett befindet, markiert. Aus diesen Punkten werden, wie bei der Grauwertkamera, die intrinsischen Parameter α_x^d , α_y^d , x_0^d , y_0^d und die extrinsischen Parameter \mathbf{R}^d und \mathbf{t}^d der Tiefenkamera bestimmt. Im Anschluss wird eine Schätzung der Tiefe Z_{cam}^d erstellt. Diese wird genutzt, um, mit der inversen Disparität d aus Gleichung 3.17, ein lineares Gleichungssystem zu erstellen. Die Lösung des Gleichungssystems ergibt eine erste ungenaue Schätzung der Parameter β und γ .

Die relative Position der Kameras zueinander wird durch \mathbf{R}^r und \mathbf{t}^r beschrieben. Jedoch kann diese nicht, wie zuvor in Gleichung 3.16, direkt mit \mathbf{R} , \mathbf{t} und \mathbf{R}^d , \mathbf{t}^d kombiniert werden. Dies liegt an den unterschiedlichen Bezugssystemen (Weltkoordinatensystemen), welche ihren Ursprung in $\mathbf{0}$ bzw. $\mathbf{0}_d$ (siehe Abbildung 3.16) haben. Durch das Anbringen des Schachbretts auf einer Platte (siehe Abbildung 3.15) sind die Punkte in beiden Systemen jedoch koplanar zueinander. Dies lässt sich mit der Ebenengleichung $\mathbf{n}^T \mathbf{x} - \delta = 0$ beschreiben. \mathbf{n} ist dabei der Normalenvektor auf der Ebene und δ der Abstand zum Ursprung ($\mathbf{0}$ bzw. $\mathbf{0}_d$). In beiden Systemen gilt $\mathbf{n} = [0 \ 0 \ 1]^T$ und $\delta = 0$. Um die Ebenenparameter zwischen den beiden Systemen unterscheiden zu können, verwenden wir \mathbf{n}^d , δ^d für das Tiefenkamerasystem bzw. \mathbf{n} , δ für das System der Grauwertkamera. Durch Transformation der Parameter in die jeweiligen Kamerakoordinatensysteme, wobei die Rotationsmatrizen

$\mathbf{R} = [\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{r}_3]$ bzw. $\mathbf{R}^d = [\mathbf{r}_1^d \ \mathbf{r}_2^d \ \mathbf{r}_3^d]$ in ihre Spalten aufgeteilt werden, ergibt sich

$$\mathbf{n} = \mathbf{r}_3 \quad \text{und} \quad \delta = \mathbf{r}_3^T \mathbf{t} \quad (3.18)$$

bzw.

$$\mathbf{n}^d = \mathbf{r}_3^d \quad \text{und} \quad \delta^d = \mathbf{r}_3^{dT} \mathbf{t}^d \quad (3.19)$$

für die Tiefenkamera. Durch das Zusammenstellen der Parameter aus den einzelnen Kalibrierenaufnahmen in Matrizen $\mathbf{M} = [\mathbf{n}_1 \ \mathbf{n}_2 \ \dots \ \mathbf{n}_n]$ und $\mathbf{b} = [\delta_1 \ \delta_2 \ \dots \ \delta_n]$ bzw. $\mathbf{M}^d = [\mathbf{n}_1^d \ \mathbf{n}_2^d \ \dots \ \mathbf{n}_n^d]$ und $\mathbf{b}^d = [\delta_1^d \ \delta_2^d \ \dots \ \delta_n^d]$, lässt sich die relative Position nach Herrera u. a. [2011] aus

$$\mathbf{R}^r = \mathbf{M}^d \mathbf{M}^T \quad \text{und} \quad \mathbf{t}^r = (\mathbf{M} \mathbf{M}^T)^{-1} \mathbf{M} (\mathbf{b} - \mathbf{b}^d)^T \quad (3.20)$$

berechnen. Die so gewonnene Rotationsmatrizen \mathbf{R} bzw. \mathbf{R}^d sind durch Rauschen nicht orthonormal. Dies wird durch eine Singulärwertzerlegung korrigiert, indem $\mathbf{R} = \mathbf{U} \mathbf{S} \mathbf{V}^T$ bzw. $\mathbf{R}^d = \mathbf{U}^d \mathbf{S}^d \mathbf{V}^{dT}$ in ihre Singulärwertmatrizen zerlegt werden und anschließend durch $\mathbf{R} = \mathbf{U} \mathbf{V}^T$ bzw. $\mathbf{R}^d = \mathbf{U}^d \mathbf{V}^{dT}$ neu berechnet werden (\mathbf{S} bzw. \mathbf{S}^d werden als Einheitsmatrizen angenommen).

Der zweite Schritt aus Herrera u. a. [2011] sieht vor, eine Fehlerfunktion ähnlich der Funktion 3.15 aufzustellen. Jedoch werden für die Tiefenkamera die Differenz der gemessenen Disparität d und der, aus der Invertierung von Gleichung 3.17 berechneten, Disparität \hat{d} verwendet. Da beide Systeme unterschiedliche Einheiten aufweisen, werden sie mit der inversen Messvarianz σ^2 bzw. σ^{d^2} gewichtet. Die Fehlerfunktion

$$\begin{aligned} & \sigma^{-2} \sum_{n=1}^N \sum_{m=1}^M \|\mathbf{x}_{nm} - \hat{\mathbf{x}}(\alpha_x, \alpha_y, x_0, y_0, \kappa_1, \kappa_2, \kappa_3, \kappa_4, \mathbf{R}_n, \mathbf{t}_n, \mathbf{X}_m)\|^2 + \\ & \sigma^{d-2} \sum_{n=1}^N \sum_{m=1}^M \|d_{nm} - \hat{d}(\alpha_x^d, \alpha_y^d, x_0^d, y_0^d, \beta, \gamma, \mathbf{R}_n^d, \mathbf{t}_n^d, \mathbf{X}_m^d)\|^2 \end{aligned} \quad (3.21)$$

wird dann durch ein nichtlineares Optimierungsverfahren (Levenberg-Marquardt Algorithmus [Hartley u. Zisserman 2004]) minimiert. Für die Tiefenkamera wird dabei keine Verzerrung angenommen ($\kappa_1^d = \kappa_2^d = \kappa_3^d = \kappa_4^d = 0$). Bei der Grauwertkamera wird, zusätzlich zur radialen Verzerrung (κ_1, κ_2), die tangentielle Verzerrung (κ_3, κ_4) berücksichtigt. Um der zuvor gemachten Schätzung der Tiefenkameraparameter gerecht zu werden, welche im Vergleich zu den Parametern der Grauwertkamera sehr ungenau sind, wird die nichtlineare Optimierung in einer ersten Phase mit festen Grauwertkameraparametern durchgeführt. In der zweiten Phase werden dann alle Parameter optimiert.

3.4. Detektion

Ohne vorherige Information zu Zeichnungen der Tiere können diese nur schwer in Grauwertbildern aufgefunden werden. Durch die hohe Vielzahl an Variationen in der Zeichnung und zum Teil sehr kontrastreichen Übergängen in der Fellfarbe ist eine zuverlässige Detektion von Landmarken oder Formen anhand von 2D Bilddaten kaum möglich. Daher wird die Detektion der Pferdeköpfe in der vorliegenden Arbeit auf Basis der Tiefeninformation durchgeführt.

Das Segmentieren entspricht einer Zuordnung der Pixel im Bild in einzelne Teilmengen. Die Zuordnung der Pixel wird durch deren Eigenschaften (hier die Ebenenparameter aus Kapitel 3.4.1) bzw. Eigenschaften in einer Nachbarschaft bestimmt. Die Teilmengen sind beispielsweise Vorder- und Hintergrund. Die Segmentierung erzeugt disjunkte Teilmengen, so dass ein Pixel nur einer Teilmenge zugeordnet wird. Ist die Segmentierung vollständig, so sind alle Pixel des Bildes einer Teilmenge zugeordnet. Auf dem segmentierten Bild sind die Teilmengen als Regionen darstellbar.

Das Clustern dient dazu eine Gruppierung anhand einer Gewichtung oder Distanz innerhalb einer Datenmenge ausfindig zu machen. Beim Clustern wird meist eine größere Anzahl an Parametern zur Gruppierung verwendet. Die Datensätze müssen dabei nicht disjunkt einer Gruppe zugeordnet werden, so dass ein Element zwei Clustern zugeordnet sein kann oder eine hierarchische Struktur zwischen den Gruppen aufgebaut wird. Das Verfahren aus Kapitel 3.4.2 wendet nach dem eigentlichen Clustern die *Regionenmarkierung* an, um entsprechend zweideutige Zuordnungen zu entfernen. So erzeugt das im Kapitel 3.4.2 beschriebene Verfahren disjunkte Regionen im Bild, wodurch das Ergebnis mit dem einer Segmentierung vergleichbar ist.

Das Detektieren hingegen beschreibt die Zuordnung von Klassen mittels eines Klassifikators. Dabei bestimmt der Klassifikator nach den Eigenschaften des zu untersuchenden Objekts die Klasse. Häufig wird ein einfacher Schwellwert angewendet. Im einfachsten Fall können einzelne Segmente beispielsweise in die Klassen „gut“ und „schlecht“ unterteilt werden. So dass durch ein Unterscheidungskriterium zwei Klassen unterschieden werden. Die Detektion steht dabei immer im Zusammenhang mit den zu erkennenden Klassen. Das Segmentieren kann ebenfalls eine Art der Detektion sein (Beispiel Schwellwertverfahren). Da diese ein Bild anhand seiner Eigenschaften in unterschiedliche Klassen, die einzelnen Segmente, aufteilt. Die Anzahl der Klassen ist hier meist variable und bezieht sich auf Regionen im Bild. Zudem wird in der Bildverarbeitung die Segmentierung oft als ein Teilschritt der Detektion verwendet. Es handelt sich dabei jedoch um unterschiedliche Klassifikationen, so dass auch von einer Anreihung von Detektionen gesprochen werden kann. Entsprechendes findet sich auch in diesem Kapitel, den Verfahren aus Kapitel 3.4.2 bzw. 3.4.3 zum Clustern bzw. Segmentieren folgt eine Detektion der Pferdekopfkandidaten in Kapitel 3.4.4.

3.4.1. Ebenenparameter in 3D Daten zur Segmentierung

Das Segmentieren der Pixel in dem Disparitätenbild \mathcal{I}_D , welches zur Tiefeninformation (siehe Tiefenbild \mathcal{I}_Z in Kapitel 3.3.2 auf Seite 61) umgerechnet werden kann, beschreibt die Gruppierung der Pixel nach gewissen Eigenschaften. Dieses Kapitel beschreibt, wie nach zusammenhängenden Flächen in dem Disparitätenbild gruppiert wird. Das Unterteilen der Szene in einzelne Flächen kann genutzt werden um den Szeneninhalte darzustellen [Bartoli 2007; Prankl u. a. 2013]. Es erlaubt dem Rechner die Szene in einzelne Objekte zu unterteilen, um ein Verständnis der Szene aufzubauen.

Als Beispiel erlaubt ein solches Verständnis seiner Umgebung einem Roboter innerhalb eines Gebäudes, welches sich in den meisten Fällen aus ebenen Wänden und Böden zusammensetzt, zu navigieren Biswas u. Veloso [2012] (Treppen steigen: Okada u. a. [2001]). Je nach dem wie eng die Bedingung der Planarität gewählt wird können auch Objekte wie Menschen oder Autos als eine Ebene dargestellt werden. Zudem können entsprechende nicht planare Objekte durch die individuelle Komposition mehrerer Ebenen dargestellt werden (Objekt modelliert aus mehreren Ebenen [Bartoli 2007; Prankl u. a. 2013]). Sind diese Ebenen sehr klein, können die Normalen auf den Ebenen als die Normale auf der Oberfläche des aufgenommenen Objekts interpretiert werden.

Parameterbestimmung der Oberflächenebenen

In den Arbeiten von Trucco u. a. [2003]; Thakoor u. Jung [2007]; Chumerin u. van Hulle [2008] finden sich Ansätze zur Detektion von Ebenen, welche direkt aus den Disparitäten bestimmt werden. In der hier vorliegenden Arbeit werden die Disparitäten direkt von der Tiefenkamera Kinect aus Kapitel 3.3 ausgegeben. Die aufgeführten Arbeiten waren darauf angewiesen, die Disparitäten beispielsweise durch Korrespondenzenfindung aus Stereo-Bildern zu bestimmen.

Die Ebene im \mathbb{R}^3 ist durch die Gleichung

$$\Pi_1 X + \Pi_2 Y + \Pi_3 Z + \Pi_4 = 0 \quad (3.22)$$

beschrieben [Hartley u. Zisserman 2004]. $\mathbf{X} = [X \ Y \ Z]^T$ sind dabei die Punkte im Raum \mathbb{R}^3 auf der Ebene Π mit den Parametern Π_i , $i = 1 \dots 4$. Der Normalenvektor $\mathbf{n} = [X_N \ Y_N \ Z_N]^T$ der Ebene kann aus den Parametern der Ebene $\mathbf{n} = \frac{1}{\sqrt{\Pi_1^2 + \Pi_2^2 + \Pi_3^2}} [\Pi_1 \ \Pi_2 \ \Pi_3]^T$ bestimmt werden. $\delta = \frac{\Pi_4}{\sqrt{\Pi_1^2 + \Pi_2^2 + \Pi_3^2}}$ beschreibt den kleinsten Abstand der Ebene zum Ursprung des Koordinatensystems.

Um die Ebenengleichung direkt auf die Disparitäten anzuwenden, wird die Gleichung 3.22 nach dem im Kapitel 3.3.2 beschriebenen Modell angepasst. Entsprechend wurde in den Arbeiten [Thakoor u. Jung 2007; Chumerin u. van Hulle 2008] mit einem vereinfachten Modell vorgegangen. Da keine Verzerrung und extrinsi-

3. Material und Methoden

sche Transformation der Tiefenkamera angenommen wird, ergibt sich nach der Gleichung 3.5 durch die Projektion von $\mathbf{X} = [X \ Y \ Z]^T$ der Pixel $\mathbf{x} = [u \ v]^T$ mit $u = \frac{\alpha_x X}{Z} + x_0$ und $v = \frac{\alpha_y Y}{Z} + y_0$. s ist hier $s = 0$, da die Kalibrierung der Kameras nach Herrera u. a. [2011] keine Scherung vorsieht. Die Tiefeninformation Z wird nach der Gleichung 3.17 durch $Z = \frac{1}{\beta(d-\gamma)}$ aus der Disparität $d = \mathcal{I}_D(u, v)$ in dem Pixel $\mathbf{x} = [u \ v]^T$ des Disparitätenbilds \mathcal{I}_D berechnet. Die Substitution von

$$X = \frac{Z}{\alpha_x} (u - x_0), \quad Y = \frac{Z}{\alpha_y} (v - y_0) \quad \text{und} \quad Z = \frac{1}{\beta(d - \gamma)}, \quad (3.23)$$

die die Koordinaten der Punkte \mathbf{X} im \mathbb{R}^3 beschrieben durch den Inhalt des Disparitätenbilds sind, in Gleichung 3.22 ergibt

$$\Pi_1 \frac{u - x_0}{\alpha_x \beta(d - \gamma)} + \Pi_2 \frac{v - y_0}{\alpha_y \beta(d - \gamma)} + \Pi_3 \frac{1}{\beta(d - \gamma)} + \Pi_4 = 0. \quad (3.24)$$

Aufgelöst nach der Disparität d

$$d = -\frac{\Pi_1}{\beta \Pi_4 \alpha_x} u - \frac{\Pi_2}{\beta \Pi_4 \alpha_y} v - \frac{\Pi_3}{\beta \Pi_4} + \gamma + \frac{\Pi_1}{\beta \Pi_4 \alpha_x} x_0 + \frac{\Pi_2}{\beta \Pi_4 \alpha_y} y_0 \quad (3.25)$$

ergibt sich durch

$$d = \pi_1 u + \pi_2 v + \pi_3 \quad (3.26)$$

mit

$$\pi_1 = -\frac{\Pi_1}{\beta \Pi_4 \alpha_x}, \quad \pi_2 = -\frac{\Pi_2}{\beta \Pi_4 \alpha_y} \quad \text{und} \quad \pi_3 = \frac{\frac{\Pi_1}{\alpha_x} x_0 + \frac{\Pi_2}{\alpha_y} y_0 - \Pi_3}{\beta \Pi_4} + \gamma \quad (3.27)$$

die Gleichung zum Aufstellen eines linearen Gleichungssystems.

Zur Bestimmung des Normalenvektors \mathbf{n} auf der in dem Pixel $\mathbf{x} = [u \ v]^T$ abgebildeten Oberfläche der Szene, wird um den Pixel \mathbf{x} eine Region von Pixeln $\mathbf{x}_{i,j} = [u + i \ v + j]^T$, mit $i, j \in [-m, m]$ berücksichtigt. Diese Region $\mathbf{x}_{i,j}$ um den Pixel \mathbf{x} wird im Folgenden auch als Patch bezeichnet.

Das lineare Gleichungssystem wird mit der Gleichung 3.26 aus den Pixeln $\mathbf{x}_{i,j}$ des Patches um \mathbf{x} mit der Breite $2m + 1$ aufgestellt. Ignoriert werden die durch zu geringe Entfernung oder Verdeckung ungültigen Pixel, die im Disparitätenbild eindeutig zu erkennen sind. Sind in dem Patch mehr als drei gültige Pixel, so hat das Gleichungssystem mehr Gleichungen als Unbekannte und es wird überbestimmt. Die meist überbestimmten Gleichungssysteme werden als Ausgleichsproblem gelöst. Nach dem Lösen des Gleichungssystems sind die Ebenen in Form der Disparitäten-Parameter π_i , $i = 1 \dots 3$ verfügbar. Diese sind nun zurück in die Parameter der Ebene Π im \mathbb{R}^3 zu transformieren.

Die Parameter Π_i , $i = 1 \dots 3$ lassen sich durch Umformung von 3.27 aufstellen:

$$\begin{aligned}\Pi_1 &= -\beta \Pi_4 \alpha_x \pi_1, \quad \Pi_2 = -\beta \Pi_4 \alpha_y \pi_2 \quad \text{und} \\ \Pi_3 &= -\beta \Pi_4 (\pi_3 - \gamma + x_0 \pi_1 + y_0 \pi_2) .\end{aligned}\tag{3.28}$$

Zur Bestimmung von Π_4 wird die zusätzliche Bedingung aufgestellt, dass der Normalenvektor $\mathbf{n} = [\Pi_1 \quad \Pi_2 \quad \Pi_3]^T$ die Länge 1 hat:

$$\|\mathbf{n}\| = 1 = \sqrt{\Pi_1^2 + \Pi_2^2 + \Pi_3^2} .\tag{3.29}$$

Hierdurch lässt sich Π_4 berechnen

$$\Pi_4 = \frac{1}{\beta \sqrt{\alpha_x^2 \pi_1^2 + \alpha_y^2 \pi_2^2 + (\pi_3 - \gamma + x_0 \pi_1 + y_0 \pi_2)^2}} .\tag{3.30}$$

Und Π_i , $i = 1 \dots 3$ ergeben sich zu

$$\begin{aligned}\Pi_1 &= -\frac{\alpha_x \pi_1}{\sqrt{\alpha_x^2 \pi_1^2 + \alpha_y^2 \pi_2^2 + (\pi_3 - \gamma + x_0 \pi_1 + y_0 \pi_2)^2}}, \\ \Pi_2 &= -\frac{\alpha_y \pi_2}{\sqrt{\alpha_x^2 \pi_1^2 + \alpha_y^2 \pi_2^2 + (\pi_3 - \gamma + x_0 \pi_1 + y_0 \pi_2)^2}} \\ \text{und } \Pi_3 &= -\frac{\pi_3 - \gamma + x_0 \pi_1 + y_0 \pi_2}{\sqrt{\alpha_x^2 \pi_1^2 + \alpha_y^2 \pi_2^2 + (\pi_3 - \gamma + x_0 \pi_1 + y_0 \pi_2)^2}} .\end{aligned}\tag{3.31}$$

Die Parameter Π_i , $i = 1 \dots 4$ beschreiben die Ebenen im \mathbb{R}^3 durch den Normalenvektor $\mathbf{n} = [\Pi_1 \quad \Pi_2 \quad \Pi_3]^T$ und die Distanz $\delta = \Pi_4$ vollständig und sind direkt aus den Disparitäten abzuleiten. Ein alternativer Ansatz wäre die Rekonstruktion der Raumpunkte aus den Disparitäten (siehe Anhang C), um mit den so berechneten Punkten \mathbf{X} und der Gleichung 3.22 die Parameter Π_i , $i = 1 \dots 4$ zu bestimmen. Jedoch konnte wie auch in der Arbeit von Chumerin u. van Hulle [2008] beobachtet werden, dass dieser Ansatz weniger stabil ist.

Transformation der Ebenenparameter

Die im vorherigen Kapitel beschriebenen Parameter Π_i , $i = 1 \dots 4$ der Ebenen werden für jeden Pixel einzeln bestimmt, um zusammengehörige Punkte einer Ebene aufzufinden. Um die Anzahl der Parameter und damit den Rechenaufwand bei den folgenden Schritten zu reduzieren, wird der Normalenvektor $\mathbf{n} = [\Pi_1 \quad \Pi_2 \quad \Pi_3]^T$ durch seine sphärischen Kugelkoordinaten $\hat{\psi}$ und $\hat{\theta}$ auf der Einheitssphäre repräsentiert [Hetzl u. a. 2001]:

$$\hat{\psi} = \arctan\left(\frac{\Pi_3}{\Pi_2}\right) \quad \text{und} \quad \hat{\theta} = \arctan\left(\frac{\sqrt{\Pi_2^2 + \Pi_3^2}}{\Pi_1}\right)\tag{3.32}$$

3. Material und Methoden

Dies ist in Abbildung 3.17 dargestellt. $\hat{\psi}$ und $\hat{\theta}$ liegen im Wertebereich zwischen 0 und π .

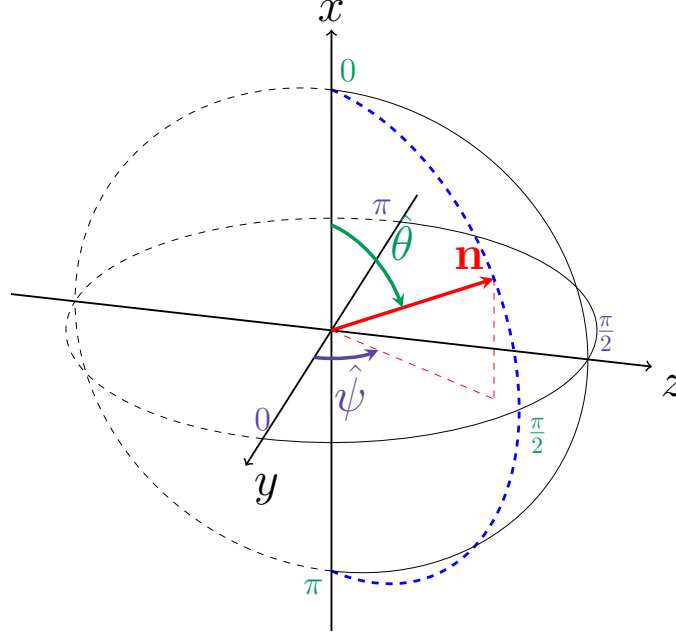


Abbildung 3.17.: Normalenvektor mit Winkeln der Einheitssphäre

Alternativ zur Gleichung 3.32 können die Winkel so bestimmt werden, dass der Normalenvektor \mathbf{n} der um den Winkel θ und ψ rotierten Z-Achse $[0 \ 0 \ 1]^T$ entspricht. Die Z-Achse wird dazu mit dem Winkel θ um die Y-Achse und mit ψ um die X-Achse rotiert, so dass sich der Normalenvektor $\mathbf{n} = [X_N \ Y_N \ Z_N]^T$ ergibt. Der Normalenvektor ist damit gleich der letzten Spalte einer Rotationsmatrix

$$\mathbf{n} = \begin{bmatrix} X_N \\ Y_N \\ Z_N \end{bmatrix} = \mathbf{R}_{\mathbf{XYZ}}(\psi, \theta, \phi) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}. \quad (3.33)$$

Mit den Rotationsmatrizen \mathbf{R}_X , \mathbf{R}_Y und \mathbf{R}_Z aus Anhang B ergibt sich durch die Multiplikation $\mathbf{R}_{\mathbf{XYZ}} = \mathbf{R}_X \mathbf{R}_Y \mathbf{R}_Z$ der Normalenvektor

$$\mathbf{n} = \begin{bmatrix} X_N \\ Y_N \\ Z_N \end{bmatrix} = \begin{bmatrix} \Pi_1 \\ \Pi_2 \\ \Pi_3 \end{bmatrix} = \begin{bmatrix} \sin(\theta) \\ -\sin(\psi) \cos(\theta) \\ \cos(\psi) \cos(\theta) \end{bmatrix}. \quad (3.34)$$

Hieraus ist ablesbar, dass $\sin(\theta) = \Pi_1$ ist. So ist

$$\tan(\theta) = \frac{\sin(\theta)}{\cos(\theta)} = \frac{\sin(\theta)}{\sqrt{1 - \sin^2(\theta)}} = \frac{\Pi_1}{\sqrt{1 - \Pi_1^2}}, \quad (3.35)$$

welches sich mit $\Pi_1^2 + \Pi_2^2 + \Pi_3^2 = 1$ (da $\|\mathbf{n}\| = 1$) zu

$$\tan(\theta) = \frac{\Pi_1}{\sqrt{1 - \Pi_1^2}} = \frac{\Pi_1}{\sqrt{\Pi_2^2 + \Pi_3^2}} \quad (3.36)$$

ergibt. Der Winkel ψ kann durch die Umformung von

$$\frac{\Pi_2}{\Pi_3} = \frac{-\sin(\psi) \cos(\theta)}{\cos(\psi) \cos(\theta)} = -\tan(\psi) \quad (3.37)$$

bestimmt werden. So dass sich durch die Anpassung für ψ und θ

$$\psi = -\arctan\left(\frac{\Pi_2}{\Pi_3}\right) \quad \text{und} \quad \theta = \arctan\left(\frac{\Pi_1}{\sqrt{\Pi_2^2 + \Pi_3^2}}\right) \quad (3.38)$$

ergibt. Diese Winkel sind entsprechend in Abbildung 3.18 gezeigt und liegen im Wertebereich zwischen $-\frac{\pi}{2}$ und $\frac{\pi}{2}$. $\hat{\psi}$ und $\hat{\theta}$ lassen sich durch

$$\psi = -\frac{\pi}{2} + \hat{\psi} \quad \text{und} \quad \theta = \frac{\pi}{2} - \hat{\theta} \quad (3.39)$$

in die Winkel ψ und θ transformieren. Denn durch $\tan(\theta) = \frac{\sin(\theta)}{\cos(\theta)} = \frac{1}{\cot(\theta)}$ lassen sich $\tan(\theta) = \cot(\hat{\theta}) = \cot(\frac{\pi}{2} - \theta)$ bzw. $\tan(-\psi) = \cot(\hat{\psi}) = \cot(\frac{\pi}{2} + \psi)$ direkt in die Transformationen aus Gleichung 3.39 überführen. Diese alternative Form die Winkel darzustellen, ermöglicht es lediglich die Absolutwerte der Winkel ψ und θ zu betrachten, da deren Wertebereich zwischen $-\frac{\pi}{2}$ und $\frac{\pi}{2}$ liegt. Dies würde zwar die Ebenen nicht mehr eindeutig darstellen, wird jedoch im Folgenden als eine alternative Methode vorgestellt.

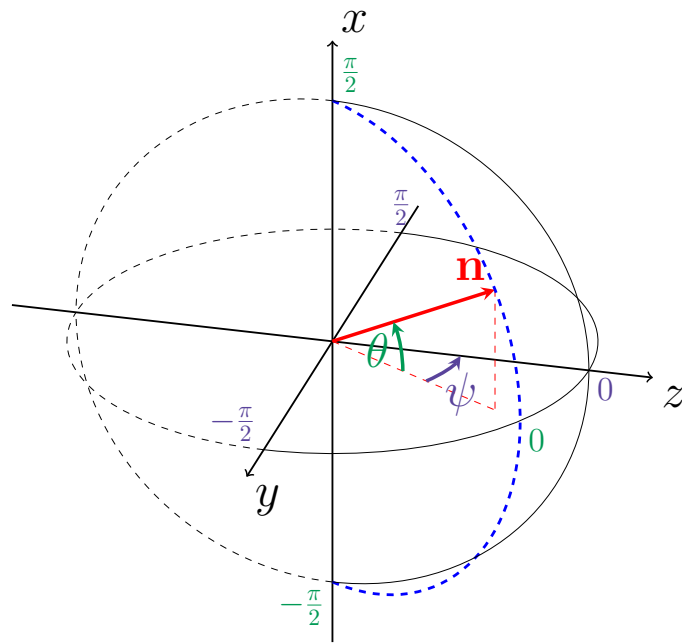


Abbildung 3.18.: Anpassung der Winkel des Normalenvektors in der Einheitssphäre

3.4.2. Clustern durch Hough-„Voting“

Die Hough-Transformation ist eine robuste Methode gegenüber Rauschen und Verdeckung. Eine Übersicht zu der Methode ist in den Arbeiten von Illingworth u. Kittler [1988]; Antolovic [2008] zu finden. Die ursprüngliche Methode von Hough [1962] dient der Liniendetektion in einem Bild. Dazu wird jeder Punkt einer Kante aus einem Binärbild in den Hough-Raum (Parameterraum) transformiert. Der Hough-Raum, auch Dualraum genannt, wird durch die Parameter zur Beschreibung einer Geraden (beispielsweise Winkel zur x-Achse und Abstand zum Ursprung) aufgespannt. In dem Dualraum entspricht ein Punkt im Originalbild einer Kurve, die alle Parameter durchläuft, welche eine Linie durch den Punkt beschreiben. Umgekehrt entspricht eine Linie im Originalbild einem Punkt im Parameterraum. In dem Dualraum werden für jeden Punkt im Originalbild die Kurven akkumuliert, daher ist dieser auch als Akkumulator bekannt. Die Punkte im Dualraum, die einen gesetzten Schwellwert überschreiten, sind die Parameter der im Originalbild detektierten Linie. Auf diese Weise lassen sich jegliche geometrische Objekte detektieren, die durch eine Ansammlung von Parametern beschrieben werden können, wie Kreise, Ellipsen oder, wie in diesem Kapitel, Ebenen.

Die Anzahl der Parameter bestimmt hierbei die Größe des Akkumulators und damit die Komplexität bzw. den Rechen- und Speicheraufwand. Die Parameter werden zudem quantisiert, der gesamte Wertebereich wird in Bereiche unterteilt. Je größer diese Quantisierungsschritte sind, desto geringer ist die Genauigkeit der durch die Hough-Transformation bestimmten Parameter. Umgekehrt steigt der Rechen- und Speicheraufwand bei kleineren Quantisierungsschritten deutlich, zudem kompensiert das Verfahren das Rauschen der Parameterwerte weniger robust.

Die Detektion der Ebenen durch die Hough-Transformation ist die Erweiterung der Liniendetektion, welche sich im \mathbb{R}^2 befinden, auf den \mathbb{R}^3 . Die Hough-Transformation sieht vor, für jeden Punkt \mathbf{X}_{Π} im Koordinatenraum alle Parameter der Ebenen im Hough-Raum zu bestimmen, auf denen der Punkt \mathbf{X}_{Π} liegt. Im Koordinatenraum ist dies die Schar aller Ebenen durch den Punkt \mathbf{X}_{Π} . Dadurch entspricht jeder Punkt des Koordinatenraumes einer gekrümmten Fläche im Parameterraum, wie es in Abbildung 3.19 dargestellt wird. Die Fläche im Hough-Raum ist die Erweiterung der zweidimensionalen Kurve der Liniendetektion, bei der drei anstelle von zwei Parametern durchlaufen werden. Für jeden Punkt im Koordinatenraum werden die entsprechenden Flächen im 3-dimensionalen Hough-Raum akkumuliert. Das Maximum im Hough-Raum entspricht den Parametern der Ebene, auf der die meisten Punkte im Koordinatenraum liegen. Dieses Verfahren ist rechenintensiv und der Speicheraufwand ist recht hoch. Zudem bedeutet jeder weitere Parameter eine weitere Dimension, die zu durchlaufen ist und den Akkumulator entsprechend vergrößert. Durch die Dualität zwischen Koordinaten- und Parameterraum entspricht jede Ebene des Koordinatensystems einem Punkt \mathbf{P}_{Π} im Hough-Raum. Die Abbildung 3.20 zeigt eine Ebene mit den Parametern \mathbf{n} (dem Normalenvektor auf der Ebene) und

3. Material und Methoden

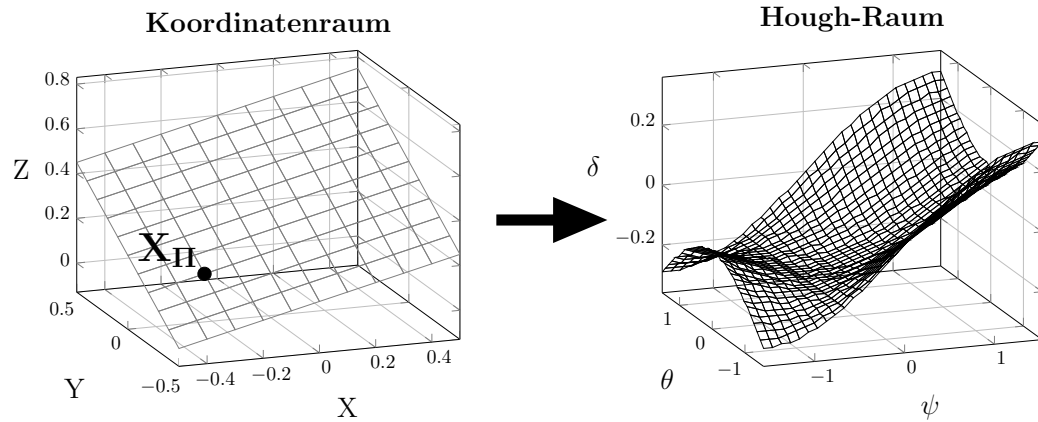


Abbildung 3.19.: Hough-Transformation für Punkt im Koordinatenraum

δ (dem Abstand der Ebene zum Ursprung) im Koordinatenraum, welche durch ihre Parameter dem Punkt \mathbf{P}_Π im Hough-Raum entspricht. Werden die Ebenenparameter wie in den Unterkapiteln „*Parameterbestimmung der Ebene*“ und „*Transformation der Ebenenparameter*“ bestimmt, können die Ebenen im Koordinatenraum durch das Hough-„Voting“ (Parameterwahl) bestimmt werden. Die Detektion der Ebenen durch die Parameterwahl ist ebenfalls in den Arbeiten von Okada u. a. [2001] und Vosselman u. a. [2004] beschrieben.

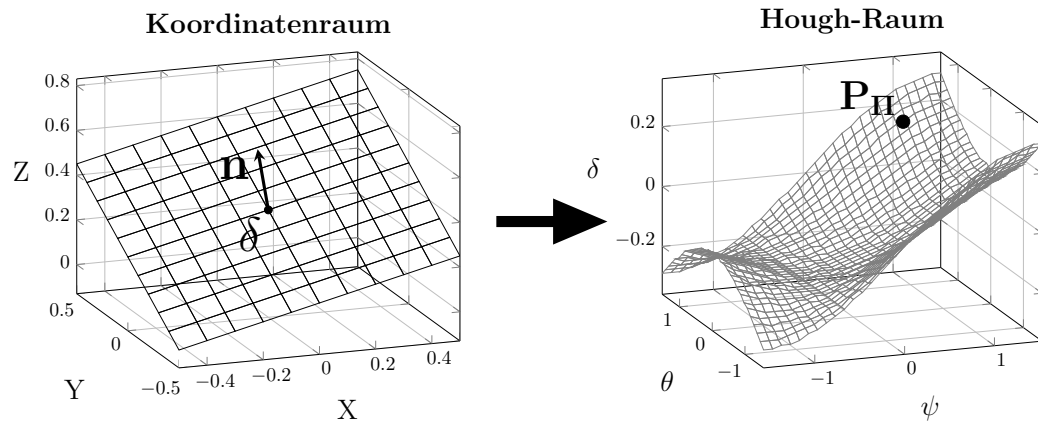


Abbildung 3.20.: Ebene mit entsprechendem Punkt im Parameterraum

Zu Anfang des Hough-„Voting“-Verfahrens wird der Akkumulator, welcher den Hough-Raum repräsentiert, auf Null gesetzt. Für jeden Pixel im Disparitätenbild werden die Parameter der Ebene über dessen Nachbarschaft (Patch) bestimmt. Anschließend wird der Akkumulator an der Stelle der Parameter des Pixels um Eins erhöht. Sind alle Pixel verarbeitet hält der Akkumulator die Anzahl der Pixel mit den quantisierten Parametern. Hierdurch entstehen Cluster (Pixel mit gleichen Eigenschaften) im Hough-Raum. Durch das Auffinden dieser Cluster, welche sich durch eine Anhäufung im Akkumulator darstellen, können die Parameter der Ebenen bestimmt werden, die einer größeren Anzahl an Pixeln entsprechen. Hierzu wird ein Schwellwert festgelegt, der ein Minimum an Pixeln für eine zu detektierende Ebene angibt.

Das Verfahren erlaubt grundsätzlich das Auffinden der gemeinsamen Eigenschaften mehrerer Pixel, in diesem Fall der Ebenenparameter. Es können in einer Szene jedoch mehrere Ebenen mit gleichen Parametern vorhanden sein, die durch ihre Anordnung im Raum keinen Zusammenhang haben. Das bis hier beschriebene Verfahren würde zwei parallele Flächen mit der selben Distanz zur Kamera, die jedoch an unterschiedlichen Positionen im Raum zu finden sind, als eine zusammengehörende Ebene erkennen. Zur Trennung dieser Flächen wird daher die im nächsten Unterkapitel beschriebene *Regionenmarkierung* eingesetzt.

Regionenmarkierung (Connected Component Labeling)

Die Zuordnung einzelner Pixel zu einer Ebene, welches im vorherigen Kapitel beschrieben wurde, kann mehrere Regionen im Bild bzw. Raum haben. Da diese meist zu unterschiedlichen Objekten gehören, ist es sinnvoll diese Gruppe von Regionen in ihre Einzelregionen zu unterteilen.

Hier setzt das Verfahren zur Regionenmarkierung an. Es wird auch in der deutschen Literatur häufig unter dem englischen Namen Connected Component Labeling behandelt. Es gibt unterschiedlichste Ausführungen, wobei dieses Kapitel auf die klassische Variante der Sequentiellen Regionenmarkierung von Rosenfeld u. Pfaltz [1966] beschränkt ist. In Stockman u. Shapiro [2001] findet sich eine detaillierte Beschreibung der in der vorliegenden Arbeit verwendeten Regionenmarkierung. Die klassische Regionenmarkierung verarbeitet das Bild zeilenweise und arbeitet auf Binärbildern. Daher wird entsprechend ein zu markierendes Binärbild für jede zuvor gefundene Ebene erzeugt, um dann mit dem Verfahren die Einzelregionen voneinander zu trennen.

Einzelne Pixel gehören dabei zu einer Region, wenn sie untereinander eine Verbindung bezüglich ihrer Nachbarschaft haben. Die hier behandelte Vierer- bzw. Achter-Nachbarschaft sind die am häufigsten auftretenden Fälle. Die direkten Nachbarpixel reichen nur in Sonderfällen nicht für das Detektieren einer Verbindung einzelner Pixel zu einer Region aus. In Abbildung 3.21 ist das Zentrumspixel \mathbf{P}_Z mit den Koordinaten (u, v) und seine Nachbarpixel in einer Vierer-Nachbarschaft dargestellt. Die Pfeile zeigen jeweils auf die Nachbarpixel $\mathbf{P}_O, \mathbf{P}_N, \mathbf{P}_W$ und \mathbf{P}_S , wobei im Algorithmus zum Unterteilen der Regionen lediglich die schraffierten Pixel \mathbf{P}_O und \mathbf{P}_N benötigt werden.

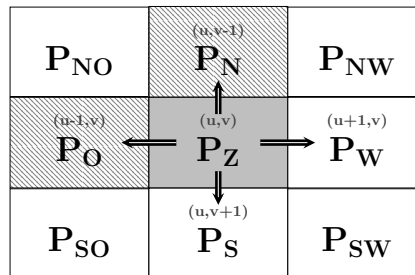


Abbildung 3.21.: Vierer-Nachbarschaft der Regionenmarkierung

Die Abbildung 3.22 zeigt wieder das Zentrumpixel \mathbf{P}_Z mit seinen Nachbarpixeln, jedoch in einer Achter-Nachbarschaft. Die Pfeile zeigen entsprechend auf die Nachbarpixel \mathbf{P}_{NO} , \mathbf{P}_N , \mathbf{P}_{NW} , \mathbf{P}_W , \mathbf{P}_{SW} , \mathbf{P}_S , \mathbf{P}_{SO} und \mathbf{P}_O . Für das Verfahren werden wieder nur die schraffierten Pixel \mathbf{P}_O , \mathbf{P}_{NO} , \mathbf{P}_N und \mathbf{P}_{NW} benötigt.

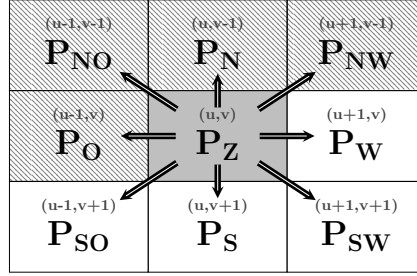


Abbildung 3.22.: Achter-Nachbarschaft der Regionenmarkierung

Der Algorithmus der Regionenmarkierung teilt sich in zwei Schritte, wobei das Bild in jedem Schritt einmal durchlaufen wird. Im ersten Schritt werden vorläufige Markierungen gesetzt, die zusätzlich in einer Vereinigungstabelle vorgehalten werden. Die Vereinigungstabelle hält dabei alle bereits gesetzten Markierungen (Label) und deren Zuordnung, die bei der Verbindung mehrerer Markierungen entstehen. Im zweiten Schritt wird die Vereinigungstabelle aufgelöst und die endgültigen Label werden im Bild gesetzt. Dazu wird die Vereinigungstabelle bis zu ihren Wurzeln aufgelöst und ergibt eine Umsetzungstabelle (LUT - Lookup Table), mit der die vorübergehend gesetzten Label in die endgültig zugeordneten Label überführt werden können. In der Literatur [Cohen 1996] wird dieses Verfahren auch als Einschritt-Verfahren (One-Pass Segmentation) bezeichnet, wobei die Umsetzungstabelle nicht in das Bild übertragen wird. Das Auflösen der Umsetzungstabelle wird hier nicht als zweiten Schritt angesehen.

Der klassische Algorithmus arbeitet sequenziell jeweils auf zwei Zeilen im Bild. Daher werden lediglich die in Abbildungen 3.21 bzw. 3.22 schraffierten Pixel der Vierer- bzw. Achter-Nachbarschaft benötigt. Im ersten Schritt wird jedes Pixel im Bild der Reihe nach als Zentrumpixel \mathbf{P}_Z behandelt. Die Reihenfolge ist dabei zeilenweise, also von oben nach unten, und in der Zeile von links nach rechts. Es werden nun drei Fälle für das Binärbild unterschieden:

1. $\mathbf{P}_Z = 0 \Rightarrow$ keine Markierung notwendig
- 2+3. $\mathbf{P}_Z = 1 \Rightarrow$ überprüfe Nachbarpixel:
 2. $\forall_{i \in \{O, NO, N, NW\}} \mathbf{P}_i = 0$, Nachbarschaft ist ohne Label
 $\Rightarrow L = L_{max} + 1$, erzeuge neues Label und
 setze es an der Position von \mathbf{P}_Z : $\mathbf{R}_{im}(\mathbf{P}_Z) = L$

3. Material und Methoden

3. $\exists_{i \in \{O, NO, N, NW\}} \mathbf{P}_i = 1$, mindestens ein Nachbarpixel ist bereits markiert
 \Rightarrow Suche kleinstes Label L_{min} aus den durch Vereinigung entstandenen
 Wurzel-Markierungen der Nachbarschaftspixel (siehe Text) und setze
 dieses an der Position von \mathbf{P}_Z : $\mathbf{R}_{im}(\mathbf{P}_Z) = L_{min}$

L_{max} stellt die Anzahl der vorübergehend vergebenen Label dar und wird zu Beginn des Verfahrens auf Null gesetzt, da noch kein Label vorhanden ist. \mathbf{R}_{im} hat die Größe des zu untersuchenden Bilds und hält alle vorübergehend gesetzten Regionen-Label. Für die Vierer-Nachbarschaft fallen lediglich die Nachbarschaftspixel \mathbf{P}_{NO} und \mathbf{P}_{NW} weg. Zu den aufgeführten Fällen ist es notwendig eine Vereinigungstabelle \mathbf{L}_V zu erstellen. Diese wird mit einem einfachen Array realisiert, in dem an der Stelle des Labels die jeweilige Vereinigung eingetragen wird. Der erste Fall, der bei einem Zentrumpixel \mathbf{P}_Z gleich Null eintritt, bedarf keiner Markierung. Im zweiten Fall, in dem das Zentrumpixel \mathbf{P}_Z der Binärmaske ungleich und alle Pixel der Nachbarschaft gleich Null sind, wird ein neues Label $L = L_{max} + 1$ erstellt. Dieses wird in die Vereinigungstabelle (einfaches Array) mit Null eingetragen $\mathbf{L}_V(L) = 0$. Die Null kennzeichnet das entsprechende Label als Wurzel, denn aus der Vereinigungstabelle kann ein Baum erstellt werden, durch den sich die Zuordnung der Label untereinander repräsentieren lässt. Der dritte Fall tritt ein, wenn neben dem Zentrumpixel \mathbf{P}_Z mindestens ein Pixel der Nachbarschaft ungleich Null ist. In diesem Fall ist es notwendig die Wurzeln der jeweiligen Label der Nachbarschaftspixel in dem Baum aufzufinden. Gestartet wird mit den Markierungen aus \mathbf{R}_{im} für alle Nachbarschaftspixel die größer Null sind $L_i = \mathbf{R}_{im}(\mathbf{P}_i) \mid \forall i \in \{O, NO, N, NW\}$ mit $\mathbf{P}_i = 1$. Anschließend erfolgt das Durchlaufen $L_i = \mathbf{L}_V(L_i)$ der Vereinigungstabelle bis $\mathbf{L}_V(L_i)$ Null ist, also die jeweilige Wurzel des Labels im Baum erreicht ist. Ist nur einer der Nachbarschaftspixel markiert, kann die entsprechende Markierung direkt übernommen werden $\mathbf{R}_{im}(\mathbf{P}_Z) = L_i$. Bei mehreren gültigen Markierungen wird das Minimum $L_{min} = \min(L_i \mid \forall i \in \{O, NO, N, NW\})$ in $\mathbf{R}_{im}(\mathbf{P}_Z) = L_{min}$ gesetzt. Zudem ist die Vereinigung in die Tabelle \mathbf{L}_V für die restlichen Label einzutragen $\mathbf{L}_V(L_i \mid \forall i \in \{O, NO, N, NW\} \mathbf{P}_i = 1 \setminus L_{min}) = L_{min}$. L_{min} ist damit die neue Wurzel aller gefundenen Label der Nachbarschaft.

Zusammenfassung und Ergänzungen des Cluster-Algorithmus

Die Einzelschritte der hier vorgestellten und in den nachfolgenden Untersuchungen eingesetzten Methode sind nochmals zusammengefasst in Abbildung 3.23 dargestellt. Das vorgestellte Verfahren bevorzugt durch das Hough-„Voting“ große Flächen. Der Hintergrund, welcher keine Pferdeköpfe beinhaltet, beeinträchtigt durch große Flächen somit das Ergebnis. Um entsprechende Bereiche des Hintergrunds ausschließen zu können, wird dieser, wie in Kapitel 3.7.2 beschrieben, zuvor angelernt. In den Fällen, in denen die Wand des V-Ausschnitts (siehe Kapitel 3.2) bewegt wurde, ist diese in den originalen Daten mit als Hintergrund angelernt worden. Daher wurde der eigentliche Hintergrund aus den Frames eines Besuchs, währenddessen sich der Hintergrund nicht verändert, neu angelernt, da die Wand durch ihre Bewegung Teil des Vordergrunds ist. Sobald die Wand nämlich heruntergefahren ist, können sich dahinter Tiere und damit Pferdeköpfe befinden. Der Hintergrund hält somit die Disparitätenwerte entsprechend der größten Entfernung zur Kamera. In den jeweils untersuchten Daten wird der Hintergrund entsprechend maskiert, so dass sich der Inhalt im Disparitätenbild \mathcal{I}_D auf den Vordergrund (VG in Abbildung 3.23) beschränkt. Hierzu gehören der Kopf und Rumpf der Tiere, sowie die sich bewegende Wand im V-Ausschnitt.

Die einzelnen Verfahren wurden für diese Arbeit in C++ mit Hilfe der Open source Computer Vision (OpenCV, Bradski [2000]) Bildverarbeitungsbibliothek eigenständig implementiert. Dies ermöglichte die entsprechende Anpassung und Kombination der Verfahren. Die einzelnen Bildverarbeitungsschritte wurden zum Teil über MEX-Funktionen in MATLAB® eingebunden, um für die Auswertung eine schnellere Anpassung der Eingangsdaten und Parameter zu gewährleisten.

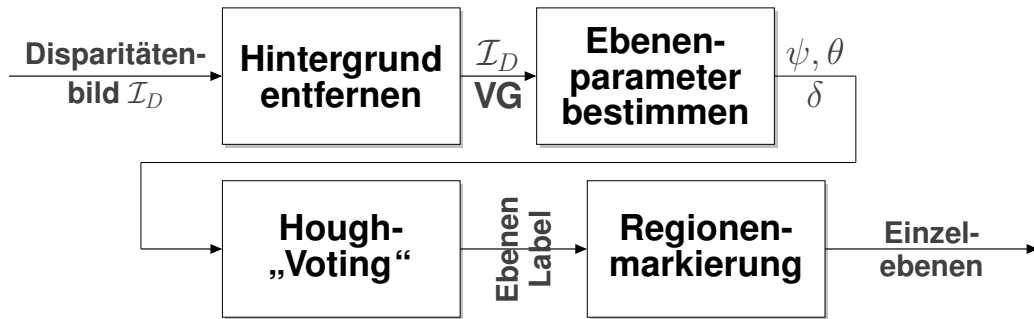


Abbildung 3.23.: Übersicht der Bildverarbeitungsschritte zum Clustern der Ebenen

Die Parameter der Ebenen ψ , θ und δ werden nicht aus einheitlich großen Patches um einen Pixel bestimmt. Statt dessen wird die Größe der berücksichtigten Region in Abhängigkeit der Disparität bestimmt. Hierdurch soll versucht werden die Größe

3. Material und Methoden

des Patches im Raum konstant zu halten, so dass die Größe des Bereiches proportional zur Disparität ist. Der umgekehrte Fall wäre in Betracht der Rauschreduktion zwar von Vorteil, jedoch werden die Patches schnell zu groß, so dass sie ausschließlich mehrere Objekte umfassen. Zur Untersuchung einzelner Objekte ist es nötig, den Patch in seiner Größe auf das Objekt zu beschränken. Sind mehrere Objekte innerhalb eines Patches, so werden die Informationen aus allen Objekten zur Bestimmung der Ebenenparameter berücksichtigt. Dies passiert ohne weitere Behandlung der Ränder vermeindlicher Objekte sowieso, jedoch nicht ausschließlich.

In Tabelle 3.3 wird die in der vorliegenden Arbeit genutzte Vorgehensweise im Vergleich zur Literatur dargestellt.

Tabelle 3.3.: Übersicht der eingesetzten Verfahren

Beschreibung: Einträge mit ● zeigen an, dass ein Verfahren in der jeweiligen Arbeit Verwendung findet. Die Markierungen ◐ bedeuten, dass hier die Ebenenparameter nicht in Bezug auf die Koordinaten im \mathbb{R}^3 bestimmt werden, sondern im Parameterraum der Disparitäten, so dass kein direkter Bezug zum \mathbb{R}^3 hergestellt wird. Die Markierung ○ in der Zeile der Arbeit von Hetzel u. a. [2001] bedeutet, dass hier weitere Angaben zur Bestimmung der Ebenenparameter fehlen, welche jedoch zumindest zu Teilen (Normalenvektor) verwendet werden.

Arbeit	Verfahren						
	Disparität	Ebenen Parameter	Iterativ	Sphärische Koordinaten	Transformation Hough	„Voting“	Regionenmarkierung
Hetzel u. a. [2001]		○		●			
Okada u. a. [2001]		●		●		●	
Trucco u. a. [2003]	●	◐				●	
Vosselman u. a. [2004]		●		●	●	●	
Thakoor u. Jung [2007]	●	◐	●				●
Chumerin u. van Hulle [2008]	●	●	●				
Eigene Arbeit	●	●		●		●	●

3.4.3. Segmentierung durch Winkel der Ebenennormalen

Die konkave Form des gesamten Tierkopfes ist in Abbildung 3.24 zu erkennen. Die Abbildung zeigt in Grün ein sagittales (3.24b) und in Rot ein transversales Profil (3.24c) eines Pferdekopfes. Beide Profile (sagittal und transversal) sind in Abbildung 3.24a auf dem Pferdekopf farblich markiert.

Die konkave Form der Pferdeköpfe ist unabhängig von der Kopfform der Einzeltiere, wie sie in Abbildung 3.25 dargestellt sind und in der Literatur [Werni 1996] unterschieden werden. Die einzelnen Kopfformen beziehen sich vorrangig auf die Form der Stirn-Nasenlinie bzw. des Mauls.

Zur Segmentierung wird in diesem Fall das Abfallen der Profile zu den Seiten hin ausgenutzt. Dieses ist, durch die konkave Form des Kopfes, aus weiten Bereichen unterschiedlicher Blickrichtungen auf den Kopf erkennbar. Sicherlich beschränkt sich diese Methode auf eine grobe Orientierung des Kopfes zur Kamera hin. Jedoch werden Köpfe, welche eine zu hohe Abweichung von der direkten Kameraufsicht haben, zur weiteren Erkennung ohnehin unbrauchbar. Eine zu hohe Abweichung führt unter anderem dazu, dass Teile des Kopfes durch sich selbst verdeckt werden.

Ziel dieses Verfahrens ist es das Tiefenbild in unterschiedliche Regionen aufzuteilen. Sollte ein Pferdekopf im Bild vorhanden sein, soll dieser ein vom Rest des Bildes unterscheidbares Label erhalten. Das einzelne Label repräsentiert dabei eine Region im Bild, welche sich durch eine Maske darstellen lässt.

3. Material und Methoden

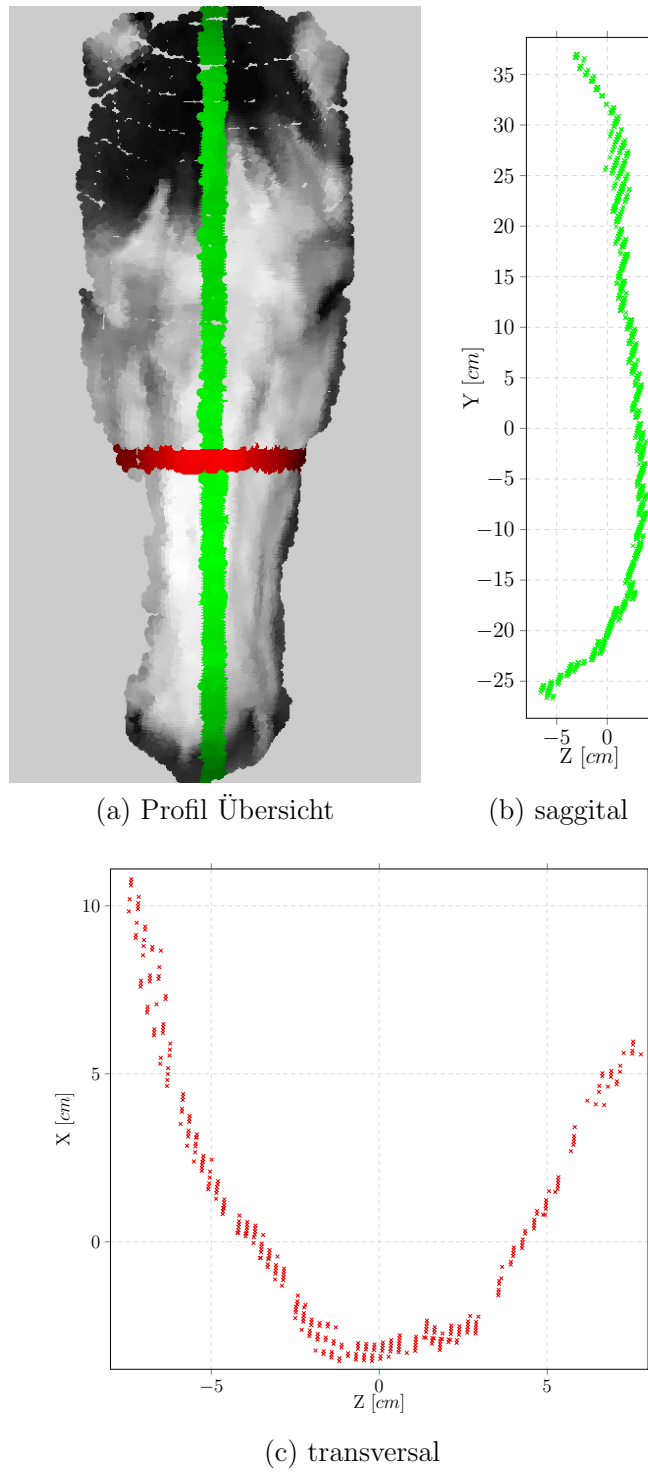
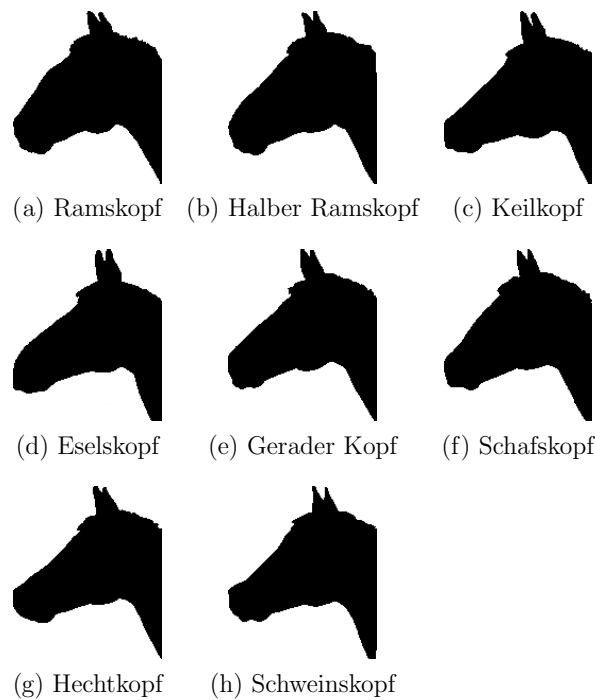


Abbildung 3.24.: Pferdekopf Profile a) Übersicht b) transversal c) sagittal



Abbildungung 3.25.: Kopfformen [Verändert nach: Werni 1996]

Segmentieren hoher Winkelabweichungen der Ebenennormalen zur Blickrichtung vom Kamerazentrum

Zur Trennung der einzelnen Regionen werden über das gesamte Bild die Randbereiche der Profile (siehe Abbildung 3.24b/c) eines Pferdekopfes gesucht. Die entsprechenden Randbereiche der Profile werden in einer Maske gehalten und erhalten, ebenso wie die Pixel, deren Nachbarschaft zur Bestimmung der Ebenenparameter Kapitel 3.4.1 ungültige Disparitätenwerte enthalten, den Wert 0. Alle verbleibenden Regionen im gesamten Bild erhalten den Wert 1, so dass die Maske lediglich ungültige und solche Regionen im Randbereich eines Profils ausschließt. Zur Bestimmung der Randbereiche der Profile werden, wie bei dem vorherigen Verfahren aus Kapitel 3.4.2, die Ebenenparameter aus Kapitel 3.4.1 für jeden Pixel des Disparitätenbilds bestimmt. Die Ebenenparameter umfassen den Normalenvektor \mathbf{n} und die Ebenendistanz δ , welche die kürzeste Entfernung der Ebene zum Ursprung darstellt. Der Ursprung liegt in diesem Fall im Kamerazentrum \mathbf{C} . Der Normalenvektor \mathbf{n} kann zudem über die Winkel ψ und θ repräsentiert werden.

In Abbildung 3.26 ist eine Aufsicht zu sehen, welche das Kamerazentrum \mathbf{C} und einen Punkt \mathbf{X} (rot) im Raum zeigt. Die Blickrichtung vom Kamerazentrum \mathbf{C} auf den Punkt \mathbf{X} entspricht der Verbindungslinie $\overline{\mathbf{C}\mathbf{X}}$ zwischen \mathbf{C} und \mathbf{X} . Zudem ist die Ebene Π und deren Normalenvektor \mathbf{n} (blau) im Raum dargestellt. Diese Ebe-

3. Material und Methoden

ne Π stellt die Orientierung der Oberfläche im Punkt \mathbf{X} dar, welche sich aus der Anordnung der Punkte um \mathbf{X} auf der Oberfläche der aufgenommenen Szene ergibt (entsprechend Kapitel 3.4.1).

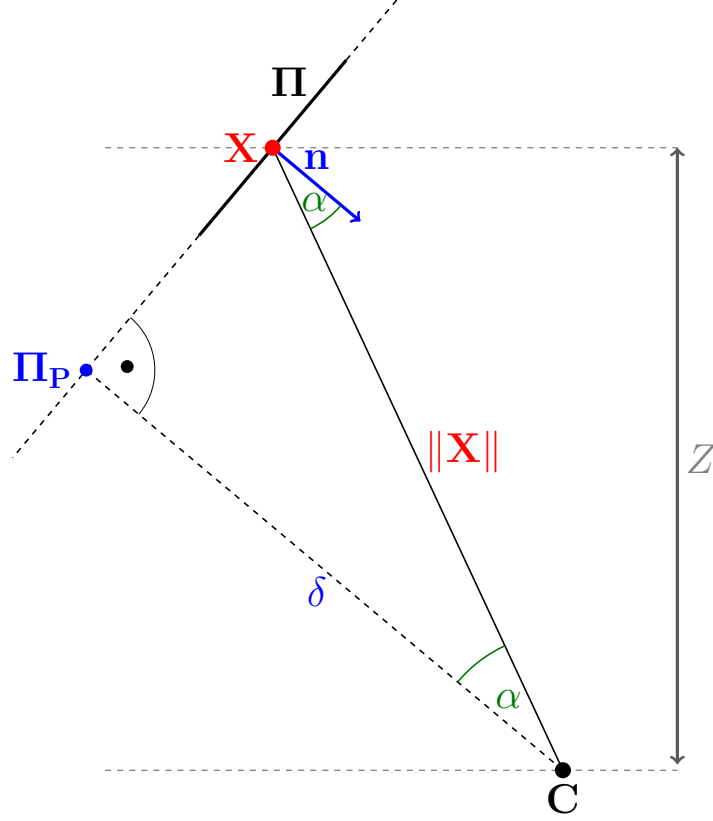


Abbildung 3.26.: Winkellabweichung der Ebenennormalen \mathbf{n} zur Blickrichtung

Die kürzeste Distanz δ der Ebene Π zum Ursprung, hier das Kamerazentrum C , wird in Abbildung 3.26 durch die Strecke $\overline{C\Pi_P}$ dargestellt. Die Verbindung $\overline{C\Pi_P}$ zwischen C und Π_P steht immer senkrecht auf der Ebene Π . Hierdurch ist das Dreieck $\overline{C\Pi_P X}$ zwischen den Punkten C , \mathbf{X} und Π_P ein rechtwinkliges.

Der Winkel α (grün) beschreibt den Winkel zwischen der Oberflächennormalen \mathbf{n} (blau) und der Strecke \overline{CX} , also der Blickrichtung vom Kamerazentrum C auf den Punkt \mathbf{X} . Der entsprechende Wechselwinkel findet sich zwischen der Strecke $\overline{C\Pi_P}$ und der Strecke \overline{CX} , dadurch dass $\overline{C\Pi_P}$ und \mathbf{n} parallel zueinander verlaufen. Der Winkel α (grün) innerhalb des rechtwinkligen Dreiecks $\overline{C\Pi_P X}$ ergibt sich durch

$$\alpha = \arccos\left(\frac{\delta}{\|\mathbf{X}\|}\right) \quad (3.40)$$

aus [Bronštejn u. a. 2012]

$$\cos(\alpha) = \frac{\delta}{\|\mathbf{X}\|} . \quad (3.41)$$

Der Abstand δ der Ebene Π wird über ein Ausgleichsproblem der Nachbarschaft um \mathbf{X} bestimmt, wodurch der Punkt \mathbf{X} nicht auf der Ebene Π liegen muss. Der Abstand $\|\mathbf{X}\|$ wird jedoch als Approximation für den tatsächlichen Abstand $\|\mathbf{X}\| + \epsilon_X$ eines Punktes auf der Ebene Π in Richtung \mathbf{X} angenommen. Dies ist in den verwendeten Daten zu vertreten, da die Distanzen $\|\mathbf{X}\|$ und δ um ein vielfaches größer sind als die Abweichung ϵ_X .

Die Länge der Strecke $\overline{\mathbf{C}\mathbf{X}}$ entspricht dabei

$$\|\mathbf{X}\| = \sqrt{X^2 + Y^2 + Z^2} . \quad (3.42)$$

Bei der Berechnung von $\|\mathbf{X}\|$ aus dem Disparitätenwert d kann ausgenutzt werden, dass lediglich die Entfernung

$$Z = \frac{1}{\beta(d - \gamma)} \quad (3.43)$$

nach einer *Kamerakalibrierung* entsprechend Gleichung 3.17 aus Kapitel 3.3.2 neu zu berechnen ist. Indem

$$X = \frac{u - x_0}{\alpha_x} Z \quad \text{und} \quad Y = \frac{v - y_0}{\alpha_y} Z \quad (3.44)$$

aus der Gleichung C.6 des Anhangs C für die Berechnung von $\|\mathbf{X}\|$ mit Gleichung 3.42 verwendet wird, ergibt sich

$$\begin{aligned} \|\mathbf{X}\| &= \sqrt{\left(\frac{u-x_0}{\alpha_x} Z\right)^2 + \left(\frac{v-y_0}{\alpha_y} Z\right)^2 + Z^2} \\ &= Z \sqrt{\left(\frac{u-x_0}{\alpha_x}\right)^2 + \left(\frac{v-y_0}{\alpha_y}\right)^2 + 1} \end{aligned} \quad (3.45)$$

Für jeden Pixel (u, v) des Disparitätenbilds $\mathcal{I}_D : (u, v) \in \mathbb{N}_0^2 \mapsto \mathcal{I}_D(u, v) := d \in \mathbb{Z}$ lässt sich durch Gleichung 3.43 ein Tiefenwert Z bestimmen, wodurch sich das Tiefenbild $\mathcal{I}_Z : (u, v) \in \mathbb{N}_0^2 \mapsto \mathcal{I}_Z(u, v) := Z \in \mathbb{R}$ ergibt (siehe Seite 61):

$$\mathcal{I}_Z(u, v) = \frac{1}{\beta(\mathcal{I}_D(u, v) - \gamma)} . \quad (3.46)$$

3. Material und Methoden

$$\mathcal{D}_X(u, v) = \sqrt{\left(\frac{u-x_0}{\alpha_x}\right)^2 + \left(\frac{v-y_0}{\alpha_y}\right)^2 + 1} \quad (3.47)$$

kann nun direkt nach der *Kamerakalibrierung* für jeden Pixel (u, v) berechnet werden. Dabei ändert sich $\mathcal{D}_X(u, v)$, bei gleichbleibender *Kamerakalibrierung*, für alle weiteren Aufnahmen nicht mehr.

Für das gesamte Bild ergibt sich mit dem Tiefenbild \mathcal{I}_Z nach Gleichung 3.45 das Entfernungsbild $\mathcal{I}_{X_D} : (u, v) \in \mathbb{N}_0^2 \mapsto \mathcal{I}_{X_D}(u, v) := \|\mathbf{X}\| \in \mathbb{N}_0$ der Punkte \mathbf{X} für jeden Pixel (u, v) , welches sich durch

$$\mathcal{I}_{X_D}(u, v) = \mathcal{I}_Z(u, v) \mathcal{D}_X(u, v) \quad (3.48)$$

berechnet.

Die Maske über das gesamte Bild, welche die Randbereiche der Profile aus Abbildung 3.24 ausschließt, wird aus Gleichung 3.41 abgeleitet. Hierzu wird ein Winkel α aus Abbildung 3.26 festgelegt, welcher der Winkelabweichung des Normalenvektors \mathbf{n} zur Blickrichtung entspricht. Die Maske wird nach der Ungleichung

$$\frac{\delta}{\|\mathbf{X}\|} < \cos(\alpha) \quad (3.49)$$

erzeugt. Da δ und $\|\mathbf{X}\|$ absolute Längen sind, steckt in dem Winkel α , welcher die Abweichung von der Blickrichtung (Verbindungsline $\overline{\mathbf{C}\mathbf{X}}$) angibt, keine Richtungsinformation. Die Werte von α liegen zwischen 0° und 90° . Mit $\mathcal{D}_E(u, v)$, welches der Abbildung $\mathcal{D}_E : (u, v) \in \mathbb{N}_0^2 \mapsto \mathcal{D}_E(u, v) := \delta \in \mathbb{R}$ entspricht durch die jedem Pixel die Ebenendistanz δ zugeordnet ist, lässt sich die Ungleichung 3.49 mit Gleichung 3.48 zu

$$\begin{aligned} \frac{\mathcal{D}_E(u, v)}{\mathcal{I}_{X_D}(u, v)} &= \frac{\mathcal{D}_E(u, v)}{\mathcal{I}_Z(u, v) \mathcal{D}_X(u, v)} < \cos(\alpha) \\ \Rightarrow \frac{\mathcal{D}_E(u, v)}{\mathcal{I}_Z(u, v)} &< \cos(\alpha) \mathcal{D}_X(u, v) \end{aligned} \quad (3.50)$$

umformen. $\cos(\alpha) \mathcal{D}_X(u, v)$ ändert sich nach einer *Kamerakalibrierung* nicht und kann zur Erstellung der Maske für alle weiteren Aufnahmen berechnet werden.

Abbildung 3.27 zeigt beispielhaft Masken, deren Werte die einzelnen Bereiche zur Trennung der Regionen halten. Auf der linken Seite (Abbildung 3.27a) ist die Disparität farbkodiert dargestellt. In Abbildung 3.27b sind weiße, hellgraue, dunkelgraue und schwarze Bereiche zu erkennen. Die weißen Bereiche sind durch Umkehr der Ungleichung 3.50 mit dem Winkel $\alpha = 60^\circ$ entstanden. Die dunkelgrauen Flächen enthalten durch die Tiefenkamera für ungültig bestimmte Pixel. Die hellgrauen Bereiche zeigen Bereiche der Nachbarschaft dieser ungültigen Pixel. Für diese Pixel wurden keine Ebenenparameter durch die im Kapitel 3.4.1 beschriebene Metho-

de berechnet, da die Nachbarschaft mindestens einen ungültigen Pixel enthält. Die schwarzen Bereiche stellen die verbleibenden Regionen im Bild dar, deren Trennung durch das in diesem Kapitel beschriebene Verfahren beschrieben ist.

In Abbildung 3.27c, auf der rechten Seite, sind die durch die Ungleichung 3.50 erzeugten Masken mit den Winkeln $\alpha = 30^\circ$ (rot), 45° (gelb), 60° (cyan) bzw. 75° (blau) dargestellt. Die schwarzen Bereiche dieser Abbildung 3.27c entsprechen den unmaskierten Bereichen, welche weiß, hell- oder dunkelgrau in Abbildung 3.27b zu erkennen sind. Die Pixel dieser Bereiche erfüllen demnach entweder nicht die Ungleichung 3.50 oder sind durch die Tiefenkamera als ungültig zurückgegeben bzw. befinden sich in deren Nachbarschaft, welche zur Normalenbestimmung (nach Kapitel 3.4.1) herangezogen wird. Für den Winkel $\alpha = 60^\circ$ entsprechen die schwarzen und blauen Bereiche den eben beschriebenen unmaskierten Bereichen.

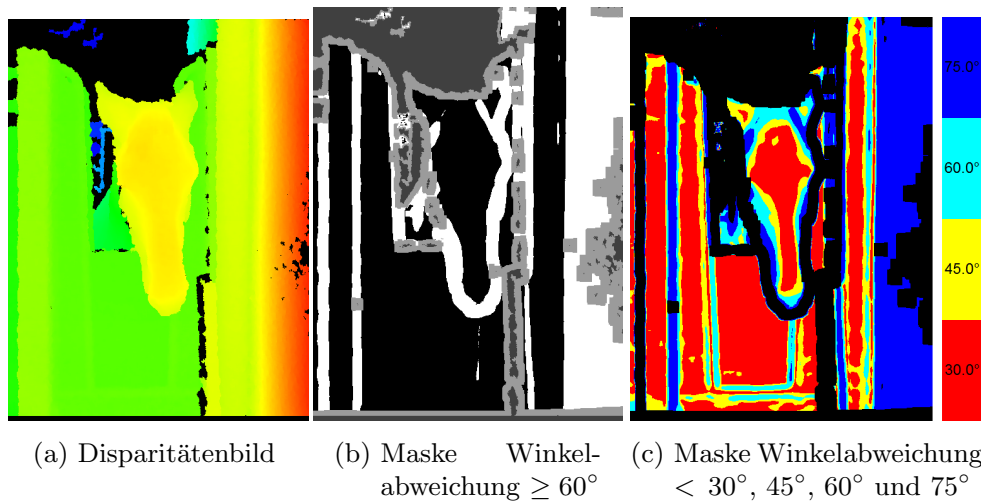


Abbildung 3.27.: Maskierung durch Winkelabweichungen

Erweiterung der Regionenmarkierung auf Mehrkanal-Bilder

Die *Regionenmarkierung* wurde bereits in Kapitel 3.4.2 auf Binärbildern beschrieben. Entsprechend dem vorherigen Unterkapitel wurde eine Maske aus den Winkelabweichungen der Normalenvektoren \mathbf{n} zur Blickrichtung erzeugt. Diese soll nun in ihre zusammenhängenden Regionen unterteilt werden. Hierzu kann der bereits verwendete Algorithmus zur *Regionenmarkierung* direkt eingesetzt werden, da es sich bei der Maske um ein Binärbild (Bild mit nur zwei Werten z. B. 0 und 1) handelt.

Abbildung 3.28 zeigt jedoch ein Beispiel, welches bei der Segmentierung der Pferdeköpfe in der gegebenen Umgebung vermehrt vorkommt. Auf der linken Seite (Abbildung 3.28a) ist das Bild der Farbkamera dargestellt, in der Mitte (Abbildung 3.28b)

3. Material und Methoden

das Disparitätenbild farbkodiert und auf der rechten Seite (Abbildung 3.28c) die entsprechenden Masken für die Winkelabweichungen $\alpha < 30^\circ$, 45° , 60° bzw. 75° . Der Kopf hat in der Abbildung nur einen geringen Abstand vom Hintergrund (Tor), so dass hier durch die im vorherigen Unterkapitel beschriebene Segmentierung keine Trennung des Pferdekopfes vom Hintergrund erfolgt. Die Nachbarschaft zur Bestimmung des Ebenenparameters δ erstreckt sich hier, durch deren geringen Abstand, über mehrere Objekte. Lediglich mit einer Winkelabweichung von $\alpha < 30^\circ$ findet eine Trennung von Tor und Kopf statt. Mit diesem Winkel wird jedoch der Kopf nicht komplett segmentiert (roter Bereich des Kopfes teilt sich in mehrere Teile). Bei den übrigen Winkeln $\alpha < 45^\circ$, 60° und 75° ist eine Verbindung des Kopfes mit dem Tor auf den Masken zu erkennen.

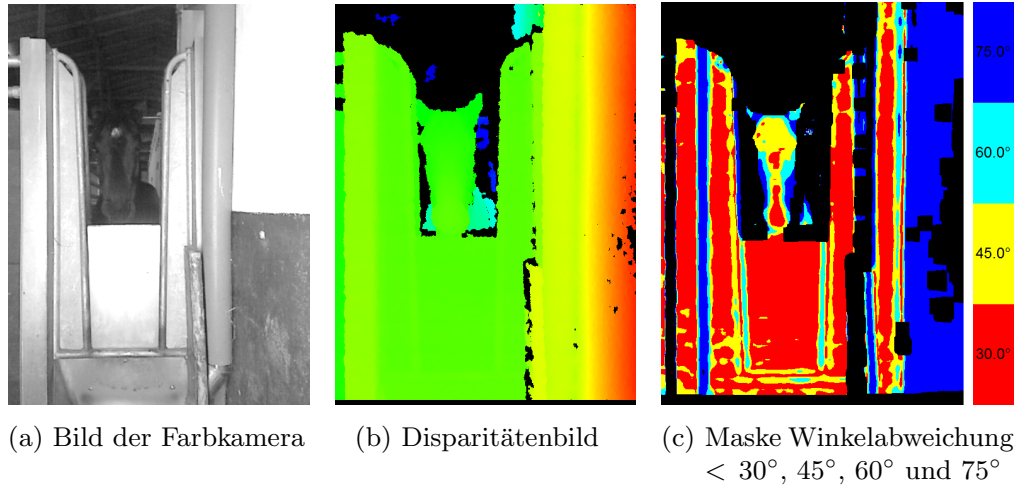


Abbildung 3.28.: Negativbeispiel: Maskierung durch Winkelabweichungen

Eine entsprechende Trennung, trotz nahe beieinanderliegender Objekte, lässt sich durch die bereits berechneten Ebenenparameter aus Kapitel 3.4.1 durchführen. Hierzu werden aus dem Disparitätenbild die Ebenenparameter ψ , θ und δ für jeden Pixel berechnet, wodurch drei Grauwertbilder entstehen. Diese lassen zu einem Bild bestehend aus mehreren Kanälen (hier bis zu drei: ψ , θ und δ) zusammenfassen. Die zuvor beschriebene *Regionenmarkierung* auf Binärbildern aus Kapitel 3.4.2 wird hierzu für das Markieren von Regionen auf Mehrkanal-Grauwertbilder erweitert.

Die drei Fälle des Algorithmus zur *Regionenmarkierung* aus Kapitel 3.4.2 bleiben. Es wird daher zunächst jedes Pixel der Maske des vorherigen Unterkapitels als Zentrumspixel \mathbf{P}_Z behandelt. Die Reihenfolge, in der die Pixel verarbeitet werden, bleibt zeilenweise und in jeder Zeile von links nach rechts. Das jeweilige Pixel der Maske \mathbf{P}_Z wird überprüft. Für den ersten Fall, dass dieses Null ($\mathbf{P}_Z = 0$) ist, ist keine Markierung notwendig, ansonsten werden die Fälle zwei und drei nach der *Regionenmarkierung* aus Kapitel 3.4.2 geprüft. Für die Fälle zwei und drei wird zur Binärmasken

ein Distanzmaß eingeführt. Mit diesem werden, zusätzlich zur Maskenbelegung der Nachbarpixel von \mathbf{P}_Z , die Grauwerte in einem Mehrkanal-Grauwertbild überprüft.

Als Distanzmaß $d^\infty(\mathbf{G}_i, \mathbf{G}_Z)$ kann nach der Maximumnorm [Bronštejn u. a. 2012, S. 284] (L_∞ -Norm) mit

$$d^\infty(\mathbf{G}_i, \mathbf{G}_Z) = \|\mathbf{G}_i - \mathbf{G}_Z\|_\infty = \max_{c \in [1 \dots N]} |G_{i,c} - G_{Z,c}| \quad (3.51)$$

verwendet werden. \mathbf{G}_i mit $i \in \{O, NO, N, NW\}$ sind die Nachbarschaftspixel von \mathbf{G}_Z auf dem Grauwertbild. \mathbf{G}_Z entspricht von der Pixelposition (u, v) dem aktuell zu verarbeitenden Zentrumpixel auf der Binärmaske \mathbf{P}_Z . Die Nachbarschaftspixel \mathbf{G}_i ($i \in \{O, NO, N, NW\}$), sowie das Zentrumpixel \mathbf{G}_Z , sind dabei Vektoren, in denen für jeden Kanal c (mit $c \in [1 \dots N]$ bei einem Bild mit N Kanälen) ein Wert $G_{i,c}$ bzw. $G_{Z,c}$ gespeichert ist. Aus dem Distanzmaß $d^\infty(\mathbf{G}_i, \mathbf{G}_Z)$ ergibt sich durch $d^\infty(\mathbf{G}_i, \mathbf{G}_Z) < T$ mit dem Schwellwert T eine Einschränkung für die Differenz der Grauwerte von \mathbf{G}_i und \mathbf{G}_Z . Die Differenz der Grauwerte liegt, für ein Bild mit zwei Kanälen, in einem Quadrat zwischen $-T$ und T .

Den Werten in den unterschiedlichen Kanälen eines Multikanal-Grauwertbildes sind häufig voneinander unabhängige Bedeutungen zugeordnet. Als Beispiel können die Werte von ψ der Ebenenparameter aus Kapitel 3.4.1 in Kanal 1 und die Werte von δ in Kanal 2 eines Zweikanal-Grauwertbildes gehalten werden. Die Werte des ersten Kanals beziehen sich somit auf einen Winkel (ψ), die von Kanal 2 auf eine Entfernung (δ). Um für die einzelnen Kanäle unterschiedliche Differenzwerte zuzulassen, kann das Distanzmaß zu

$$d^{T^\infty}(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T}) = \max_{c \in [1 \dots N]} \left| \frac{G_{i,c} - G_{Z,c}}{T_c} \right| \quad (3.52)$$

geändert werden. Der Schwellwert T_c mit $c \in [1 \dots N]$ zusammengestellt im Vektor $\mathbf{T} = [T_1 \dots T_N]^T$ gibt dabei den Bereich für jeden Kanal c einzeln an, in dem sich durch $d^{T^\infty}(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T}) < 1$ die Differenz der Werte für Kanal c bewegen dürfen $G_{i,c} - G_{Z,c} \in (-T_c, \dots, T_c)$. Die Differenz der Grauwerte liegt entsprechend, für ein Bild mit zwei Kanälen, in einem Rechteck mit den Abmessungen $2T_1 \times 2T_2$.

Das im Weiteren verwendete Distanzmaß $d^{T^2}(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T})$ basiert auf der Euklidischen Norm [Bronštejn u. a. 2012, S. 283] (L_2 -Norm). Erweitert um die Schwellwerte in \mathbf{T} ergibt sich

$$d^{T^2}(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T}) = \sqrt{\sum_{c=1}^N \frac{(G_{i,c} - G_{Z,c})^2}{T_c^2}}. \quad (3.53)$$

Mit $d^{T^2}(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T}) < 1$ wird so ein ellipsenförmiger Bereich der Differenz zwischen den Grauwerten in einem zweikanaligem Grauwertbild zugelassen.

Mit dem Distanzmaß $d^{T^2}(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T})$ wird für die Fälle zwei und drei des abge-

3. Material und Methoden

wandelten Algorithmus durch $d^{T^2}(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T}) < 1$ überprüft ob die Differenz der Grauwerte in einem vorgegebenen Bereich liegen. Wie bereits erwähnt ist dies nur notwendig, wenn der aktuell zu prüfende Pixel \mathbf{P}_Z aus der Maske des vorherigen Unterkapitels einen Wert größer Null besitzt. Denn ansonsten handelt es sich um den ersten Fall der *Regionenmarkierung* aus Kapitel 3.4.2. Die Pixel \mathbf{P}_i werden nun der Reihe nach überprüft mit $i \in \{O, NO, N, NW\}$. Ist der Wert des jeweiligen Pixels \mathbf{P}_i größer Null wird die Differenz der Grauwerte \mathbf{G}_i durch $d^{T^2}(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T}) < 1$ geprüft. Liegt die Differenz des jeweiligen Pixels \mathbf{G}_i mit \mathbf{G}_Z innerhalb der vorgegebenen Grenzen, kann nach dem alten Algorithmus zur *Regionenmarkierung* verfahren werden. Liegt die Differenz zwischen dem Pixel \mathbf{G}_i und \mathbf{G}_Z außerhalb, so wird nach dem bisherigen Algorithmus verfahren als wäre der entsprechende Pixel der Maske \mathbf{P}_i gleich Null.

Es ergibt sich der entsprechend abgeänderte Algorithmus und die Unterteilung der drei Fälle:

1. $\mathbf{P}_Z = 0 \Rightarrow$ keine Markierung notwendig
- 2+3. $\mathbf{P}_Z = 1 \Rightarrow$ überprüfe Nachbarnpixel:
 2. $\forall_{i \in \{O, NO, N, NW\}} \mathbf{P}_i = 0 \mid d(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T}) \geq 1$, Nachbarschaft ist ohne Label
 $\Rightarrow L = L_{max} + 1$, erzeuge neues Label und
 setze es an der Position von \mathbf{P}_Z : $\mathbf{R}_{im}(\mathbf{P}_Z) = L$
 3. $\exists_{i \in \{O, NO, N, NW\}} \mathbf{P}_i = 1 \ \& \ d(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T}) < 1$, mindestens ein Nachbarnpixel ist bereits markiert
 \Rightarrow Suche kleinstes Label L_{min} aus den durch Vereinigung entstandenen Wurzel-Markierungen der Nachbarschaftspixel und setze dieses an der Position von \mathbf{P}_Z : $\mathbf{R}_{im}(\mathbf{P}_Z) = L_{min}$

Entsprechend dem vorherigen Algorithmus der *Regionenmarkierung* stellt L_{max} die Anzahl der vorübergehenden Label dar, welche in \mathbf{R}_{im} für die entsprechenden Pixel gehalten werden. Die Vereinigungstabelle \mathbf{L}_V wird ebenfalls wie zuvor erstellt und gefüllt, welche beim Auffinden der Wurzel des jeweiligen Labels im Baum verwendet wird. In \mathbf{L}_V werden jedoch nur solche Pixel als Vereinigung markiert $\mathbf{L}_V(L_i \mid \forall i \in \{O, NO, N, NW\} \mathbf{P}_i = 1 \ \& \ d(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T}) < 1 \setminus L_{min}) = L_{min}$, die das zuvor beschriebene Kriterium $d^{T^2}(\mathbf{G}_i, \mathbf{G}_Z, \mathbf{T}) < 1$ erfüllen.

Ähnliche Verfahren zur Regionenmarkierung finden sich in der Literatur [Cohen 1996; Yapa u. Harada 2008; Niknam u. a. 2010], welche ebenfalls eine Anpassung zur Markierung von Grauwertbildern beschreiben. Die aufgeführten Arbeiten erstellen eine Maske zur Trennung von Vorder- und Hintergrund durch einen Schwellwert auf den Grauwertbildern. Zudem wird, ähnlich dem in diesem Kapitel beschriebenen Distanzmaß, ein Vergleich zwischen den Grauwerten der Pixel einer Nachbarschaft durchgeführt. In den Arbeiten von Cohen [1996] und Niknam u. a. [2010] findet sich

jedoch nur ein direkter Vergleich der Grauwerte. Es wird geprüft, ob es sich um identische Grauwerte in der Nachbarschaft handelt. Durch Niknam u. a. [2010] wird ein entsprechender Vergleich der Grauwerte nicht weiter beschrieben. Die in diesem Kapitel eingeführten Distanzmaße erlauben hingegen den Vergleich auf Mehrkanalbildern, sowie eine Anpassung der zulässigen Unterschiede zwischen den Grauwerten.

Mit der auf den Bildern abgebildeten Größe eines Objekts, aufgenommen in unterschiedlicher Entfernung, verändern sich auch die Differenzen zwischen den Grauwerten benachbarter Pixel. Um die Schwellwerte, welche durch die Ebenenparametern physikalischen Größen zuzuordnen sind, nicht anpassen zu müssen, werden die Grauwerte nicht in der direkten Pixelnachbarschaft verglichen. Entsprechend der Variation der Nachbarschaftsgröße bei der Berechnung der Ebenenparameter aus Kapitel 3.4.1, werden die Grauwerte in einem der Entfernung Z des Objekts entsprechenden Pixelabstand mit dem Zentrumpixel \mathbf{G}_Z verglichen. Der Pixelabstand zum Zentrumpixel \mathbf{P}_Z ist durch einen Skalierungsfaktor an die Blockgröße bei der Berechnung der Ebenenparameter gekoppelt.

Die Abbildung 3.29 zeigt eine überlagerte Darstellung von Binärmaske und Grauwertbild. Die Grauwerte sind blau gekennzeichnet und haben eine der Binärmaskenpixel gegenläufige Schraffierung. Der Grauwert \mathbf{G}_Z ist nicht gesondert dargestellt, er liegt auf der Position von Pixel \mathbf{P}_Z . Alle Grauwertpixel \mathbf{G}_{NO} , \mathbf{G}_N , \mathbf{G}_{NW} , \mathbf{G}_O und \mathbf{G}_Z tragen bei einem Mehrkanalbild alle den Kanälen entsprechenden Werte. In dem in der Abbildung gezeigten Beispiel haben die Grauwerte einen Pixelabstand von 3.

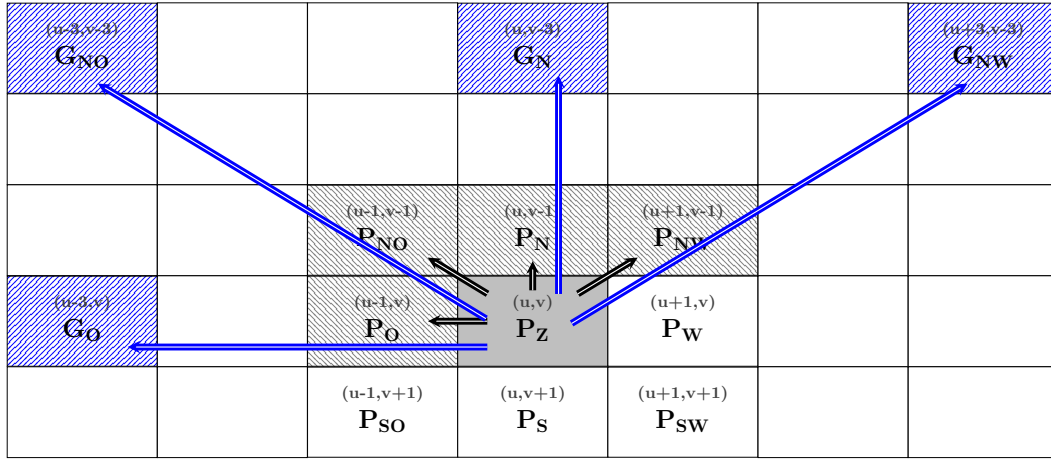


Abbildung 3.29.: Nachbarschaft der Regionenmarkierung in Grauwertbild mit vergrößertem Pixelabstand

3.4.4. Detektion von Kopfkandidaten aus segmentierten Bilddaten

Die in den Kapiteln 3.4.2 und 3.4.3 beschriebenen Verfahren bestimmen einzelne Regionen im Bild, deren Segmente über die Ebenenparameter aus Kapitel 3.4.1 getrennt wurden. Aufgabe der in diesem Kapitel beschriebenen Detektion ist es solche Segmente aus den einzelnen Segmenten im Bild zu finden, die einem Pferdekopf am ähnlichsten sind bzw. im verfolgten Fall einem Kopf entsprechen.

Zur Detektion der Pferdeköpfe werden Histogramme über ein Mehrkanalbild erzeugt. Die einzelnen Kanäle des verwendeten Mehrkanalbilds halten die bereits zur Segmentierung verwendeten Ebenenparametern ψ , θ oder δ aus Kapitel 3.4.1. Die Anzahl der Kanäle bzw. die Auswahl der Parameter kann variieren, so dass Histogramme mit einer, zwei oder drei Dimensionen entstehen. Für jedes zuvor bestimmte Segment (Region aus Kapitel 3.4.2 bzw. 3.4.3) wird ein Histogramm aus dem Mehrkanalbild erzeugt. Da die einzelnen Segmente disjunkte Regionen im Bild darstellen, haben die Werte eines Segments keinen Einfluss auf ein Histogramm eines zweiten Segments. Um unabhängig von der jeweiligen Rotation (ψ und θ) bzw. Entfernung (δ) zu werden, wird für jedes Segment der Mittelwert der einzelnen Kanäle bestimmt und von den Werten des Segments abgezogen. Zur Erstellung der Histogramme wird eine Schrittweite für jeden Kanal vorgegeben, welche den gesamten Wertebereich in mehrere Unterbereiche (Histogramm-Bins) aufteilt. Anschließend wird für jeden Bin (Wertebereich) die Häufigkeit bestimmt, in der die entsprechenden Werte in dem Segment auftreten.

Ein vergleichbarer Ansatz wurde bereits durch Tang u. a. [2013] beschrieben. Die Histogramme werden als HONV (*histogram of oriented normal vectors*) beschrieben. Das dort vorgestellte Verfahren verwendet jedoch unterschiedliche Skalierungen des Tiefenbilds (Pyramidenstruktur) und bestimmt, entgegen dem hier vorgestellten Verfahren, die Normalenvektoren bzw. die Winkel der Normalenvektoren (ψ und θ) aus der direkten Nachbarschaft. So ist es notwendig das Tiefenbild zuvor mit einem Gaußfilter von Rauschen zu befreien. Die in dieser Arbeit beschriebene variierende Größe der Nachbarschaft zur Bestimmung der Ebenenparameter in Kapitel 3.4.1 umgeht eine Behandlung unterschiedlich skalierten Tiefenbilder, sowie deren vorheriges Filtern durch einen Gaußfilter.

Weiter wird durch Tang u. a. [2013] ein Feature diskutiert, welches CIHONV (*coordinate invariant histogram of normal vectors*) genannt wird. Dieses beschreibt die Berechnung des durchschnittlichen Histogrammwerts dessen Histogramm-Bin zirkulär in eine vorgegebene Position verschoben wird. Da in Tang u. a. [2013] ausschließlich die Winkel ψ und θ der Normalenvektoren behandelt werden, wird das Histogramm unabhängig von der Rotation des zu untersuchenden Objekts im Raum. Das in dieser Dissertation verwendete Verfahren sieht vor die Werte der Kanäle jedes Segments vor der Erstellung eines Histogramms vom jeweiligen Mittelwert zu befreien. Dies trägt dazu bei Unterschiede durch eine Quantisierung bei der Erstellung der Histogramme zwischen den Segmenten zu reduzieren. Eine Quantisierung entsteht dabei durch die

zuvor festgelegten Schrittweiten bei der Zuordnung der Werte in den entsprechenden Histogramm-Bin.

Das aus jedem Segment erzeugte Histogramm \mathcal{H}_S wird mit einem Histogramm-Template \mathcal{H}_T verglichen und kann so auf einen einzigen Ähnlichkeitsfaktor ρ reduziert werden. Das Histogramm-Template wird hierzu aus einem Satz von Lern-daten bestimmt, eine nähere Beschreibung hierzu findet sich in Kapitel 3.7.5. Dabei werden die Histogramm-Bins in ihrer Schrittweite nicht verändert, so dass das zu überprüfende Histogramm eines Segments \mathcal{H}_S die selbe Größe besitzt wie das Histogramm-Template \mathcal{H}_T . Jedes Histogramm (\mathcal{H}_S und \mathcal{H}_T) wird vor einem Vergleich normalisiert, indem es durch die Summe aller Einträge H_S bzw. H_T geteilt wird:

$$\tilde{\mathcal{H}}_S = \frac{\mathcal{H}_S}{\sum_{H_S \in \mathcal{H}_S} H_S} \text{ bzw. } \tilde{\mathcal{H}}_T = \frac{\mathcal{H}_T}{\sum_{H_T \in \mathcal{H}_T} H_T} . \quad (3.54)$$

Dadurch entstehen Histogramme $\tilde{\mathcal{H}}_S$ und $\tilde{\mathcal{H}}_T$, welche die prozentuale Verteilung der jeweiligen Wertebereiche in dem zu untersuchenden Segment bzw. dem Template angeben. Die Gesamtsumme der Einträge in einem Histogramm und damit die Größe eines Segments im Bild haben dadurch keinen Einfluss auf den zu bestimmenden Ähnlichkeitsfaktor ρ . Die Ähnlichkeit $\rho = \Delta(\mathcal{H}_S, \mathcal{H}_T)$ zweier Histogramme (hier \mathcal{H}_S und \mathcal{H}_T bzw. $\tilde{\mathcal{H}}_S$ und $\tilde{\mathcal{H}}_T$) wird bestimmt, indem die Histogrammeinträge \tilde{H}_S und \tilde{H}_T über die Euklidische Norm [Bronštejn u. a. 2012, S. 283] (L_2 -Norm) der Differenz $\tilde{H}_S - \tilde{H}_T$ verglichen werden:

$$\rho = \Delta(\mathcal{H}_S, \mathcal{H}_T) = \Delta(\tilde{\mathcal{H}}_S, \tilde{\mathcal{H}}_T) = \sqrt{\sum_{\tilde{H}_S \in \tilde{\mathcal{H}}_S, \tilde{H}_T \in \tilde{\mathcal{H}}_T} (\tilde{H}_S - \tilde{H}_T)^2} \quad (3.55)$$

Der generierte Ähnlichkeitsfaktor ρ von \mathcal{H}_S zu \mathcal{H}_T ersetzt damit einen Featurevektor, welcher in Tang u. a. [2013] durch eine simple Aneinanderreihung der Einträge $H_S \in \mathcal{H}_S$ eines Histogramms generiert wird. Je ähnlicher das Histogramm \mathcal{H}_S des zu untersuchenden Segments dem Histogramm-Template \mathcal{H}_T ist, desto kleiner ist der Ähnlichkeitsfaktor ρ . Entspricht das Histogramm-Template \mathcal{H}_T einem Pferdekopf, kann mit einem zuvor festgelegten Schwellwert eine Klassifikation für jedes Segment eines Bildes ausgeführt werden. Ist ρ kleiner oder gleich dem Schwellwert wird das Segment als Pferdekopf behandelt, ist es größer als der Schwellwert wird das Segment nicht als Pferdekopf (Hintergrund) angenommen.

3.5. Posenbestimmung und Normalisierung

Die im vorherigen Kapitel 3.4 aufgefundenen Pferdeköpfe bzw. deren Kandidaten werden in diesem Kapitel zum Einsatz der Identifikation aus Kapitel 3.6 vorbereitet. Dazu werden die 2D Bilddaten durch die Normalisierung aus Kapitel 3.5.3 virtuell so manipuliert, dass die Pferdeköpfe anstatt in beliebige Richtung direkt in die Kamera schauen. Hierzu ist es jedoch zuvor notwendig die Pose, der bereits aufgefundenen Pferdeköpfe (nach Kapitel 3.4), im Raum, also die Position und Orientierung, zu bestimmen. Ein erster Schritt bestimmt dazu in Kapitel 3.5.1 die grobe Ausrichtung (Orientierung) und Position des Kopfes. In einem zweiten Schritt, welcher im Kapitel 3.5.2 beschrieben ist, wird die Pose des Kopfes zunächst ein erstes Mal verfeinert und anschließend anhand aufgefundener Landmarken ein zweites Mal korrigiert.

3.5.1. Grobe Posenbestimmung

Das in diesem Kapitel vorgestellte Verfahren verwendet die Daten, welche durch die im Kapitel 3.4.3 beschriebene Segmentierung mit einer entsprechenden Detektion der Kopfkandidaten nach Kapitel 3.4.4 bestimmt wurden. Die Parameter des Verfahrens werden entsprechend der Beschreibung in Kapitel 3.7.6 ermittelt. Die Eingangsdaten des Verfahrens sind

- Das Disparitätenbild (Kapitel 3.3)
- Die Ebenenparameter (nach Kapitel 3.4.1 aus den Normalenvektoren bestimmt)
- Die disjunkten Regionen (nach der Segmentierung aus Kapitel 3.4.3)

Die einzelnen Regionen werden, bei mehreren Kopfkandidaten, einzeln und ergeben eine Maske auf dem Disparitätenbild. Die Pose der einzelnen Kopfkandidaten kann so nacheinander bestimmt werden. Abbildung 3.30a zeigt ein Beispiel der durch die Regionenmarkierung nach Kapitel 3.4.3 entstandenen Segmentierung, bei der bereits Regionen durch die Detektion von Kopfkandidaten nach Kapitel 3.4.4 ausgeschlossen wurden. Der Schwellwert, welcher zur Detektion (Kapitel 3.4.4) in dem gezeigten Beispiel eingesetzt wurde, ist vergleichsweise hoch angesetzt, so dass neben dem tatsächlichen Pferdekopf (hellblaue Region in Abbildung 3.30a) auch weitere Regionen als Kandidaten detektiert wurden. Je enger der Schwellwert zur Detektion gefasst wird, umso weniger Regionen verbleiben als Kopfkandidaten, jedoch sinkt damit zugleich die Anzahl der korrekt detektierten Köpfe.

Abbildung 3.30b zeigt die Maske der vereinzelt Region des korrekten Kopfkandidaten aus Abbildung 3.30a (hellblaue Region). Diese wurde auf das farbkodierte Disparitätenbild angewendet (Abbildung 3.30c) und zu Zwecken der Darstellung auf das Farbbild transformiert (grüner Bereich in Abbildung 3.30d).

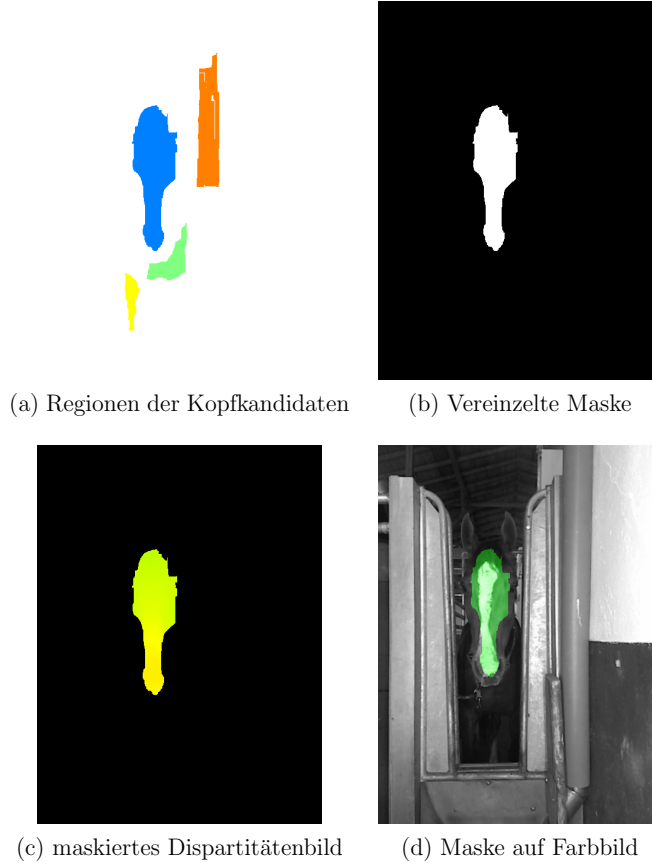


Abbildung 3.30.: Beispiel einzelnte Maske der Segmentierung nach Kapitel 3.4.3

Die grobe Position des Pferdekopfes wird zunächst auf der Maske durch deren Masseschwerpunkt \mathbf{M}_C bestimmt, dieser ist in Abbildung 3.31a als blauer Punkt auf der Maske dargestellt. Die Koordinaten X_{M_C} und Y_{M_C} des Masseschwerpunkts $\mathbf{M}_C = [X_{M_C} \ Y_{M_C}]^T$ werden nach [Bronštejn u. a. 2012, S. 197] durch

$$X_{M_C} = \frac{\sum_{i=1}^M M_i X_i}{\sum_{i=1}^M M_i} \text{ und } Y_{M_C} = \frac{\sum_{i=1}^M M_i Y_i}{\sum_{i=1}^M M_i} \quad (3.56)$$

bestimmt. M_i sind dabei die Werte der Maske, welche entweder 0 (schwarz in Abbildung 3.30b) oder 1 (weiß in Abbildung 3.30b) sind, je nachdem ob der Pixel an der entsprechenden Position (X_i, Y_i) in der Maske markiert ist. In der Gleichung 3.56 werden alle Einträge der Maske durchlaufen $i = 1 \dots M$, wobei lediglich solche Pixel Einfluss auf den Schwerpunkt haben, die auch markiert (weiß in Abbildung 3.30b) sind. Der Masseschwerpunkt hat damit die durchschnittliche Koordinate aller markierten Pixelkoordinaten in der Maske. Die besondere Form der Pferdeköpfe ähnelt einer auf den Kopf gestellten Birne, da der Kopf in Höhe der Augen breiter ist als an Nasenrücken und Nüstern. Diese Form führt dazu, dass der Masseschwerpunkt

3. Material und Methoden

sich von den Nüstern weg zur oberen „massereicheren“ Kopfhälfte bewegt.

In einem nächsten Schritt wird die Entfernung der in der Maske markierten Pixel (M_i hat den Wert 1) zum zuvor bestimmten Masseschwerpunkt \mathbf{M}_C bestimmt. Die Distanz d_{i,M_C} jedes markierten Pixels wird nach der Euklidischen Norm [Bronštejn u. a. 2012, S. 283] (L_2 -Norm) durch

$$d_{i,M_C} = \sqrt{(X_i - X_{M_C})^2 + (Y_i - Y_{M_C})^2} \quad (3.57)$$

berechnet. In Abbildung 3.31b ist die Entfernung jedes Pixels in der Maske aus Abbildung 3.31a durch die Helligkeit in Grün dargestellt. Ein helleres Grün bedeutet eine größere Distanz zum Masseschwerpunkt \mathbf{M}_C (blauer Punkt), als ein dunkleres Grün.

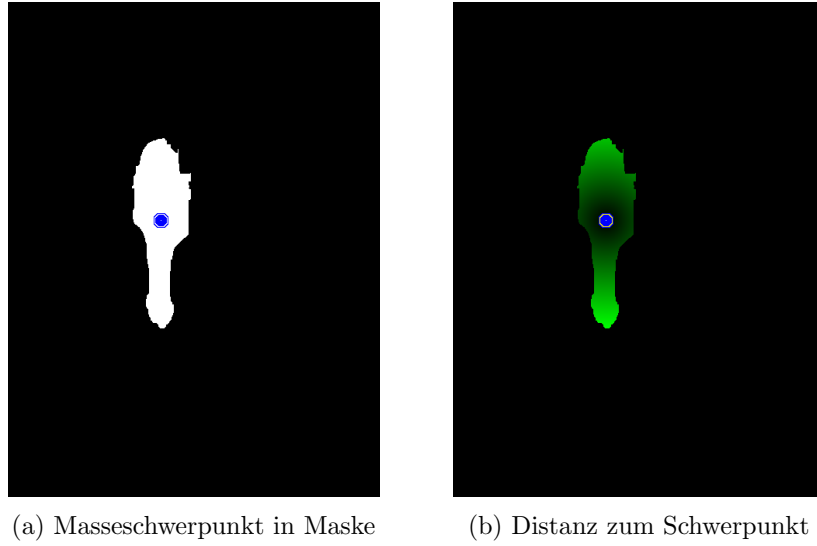


Abbildung 3.31.: Masseschwerpunkt des Kopfes auf 2D Maske

Anschließend wird ein prozentualer Schwellwert T_N festgelegt ($T_N \in [0 \dots 1]$), welcher einen Radius $R_d = T_N \max(d_{i,M_C})$ abhängig von der maximal auftretenden Entfernung $\max(d_{i,M_C})$ zum Masseschwerpunkt angibt. Alle Pixel innerhalb der Maske, deren Entfernung zum Schwerpunkt über dem bestimmten Radius $d_{i,M_C} \geq R_d$ liegen, werden als Bereich für die Nüstern weiter behandelt. Ein entsprechender Bereich mit $T_N = 0,75$, also 75% der maximal auftretenden Entfernung $\max(d_{i,M_C})$, sind für das Beispiel aus Abbildung 3.31a in Abbildung 3.32a rot dargestellt. Für die Pixel, die die Bedingung $d_{i,M_C} \geq R_d$ erfüllen, wird entsprechend der Gleichung 3.56 der Masseschwerpunkt für den Bereich der Nüstern bestimmt. Das Ergebnis entspricht dem Schwerpunkt \mathbf{M}_N , welcher in Abbildung 3.32b, neben dem Schwerpunkt der gesamten Maske (blauer Punkt), als roter Punkt dargestellt ist.

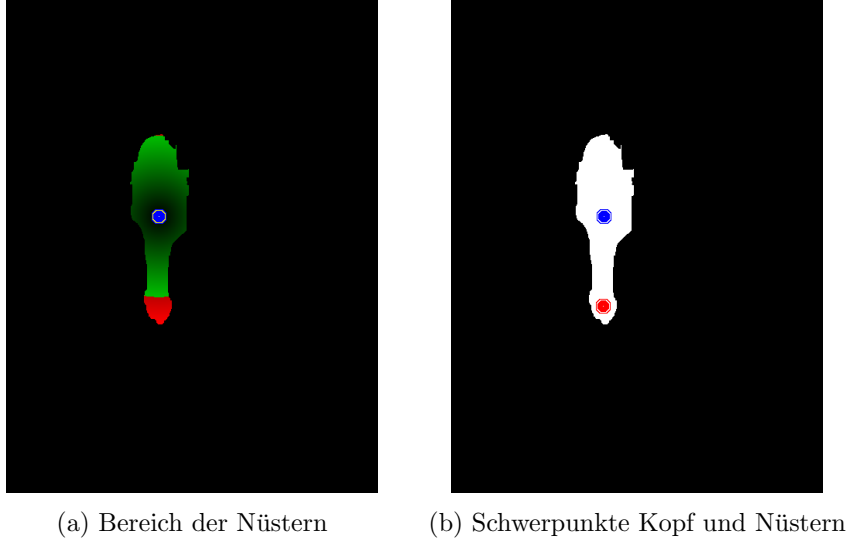


Abbildung 3.32.: Masseschwerpunkt der Nüstern auf 2D Maske

Über die beiden Masseschwerpunkte \mathbf{M}_C und \mathbf{M}_N ist eine grobe Pose des Kopfes in der 2D Maske gegeben. Die Position des Kopfes im Bild entspricht dabei dem Punkt \mathbf{M}_C . Die Orientierung wird durch den Vektor $\mathbf{V}_{CN} = \mathbf{M}_N - \mathbf{M}_C$ zwischen den beiden Punkten bestimmt.

Unter Berücksichtigung des Masseschwerpunkts \mathbf{M}_N des Bereichs der Nüstern, sowie des zur Bestimmung des entsprechenden Bereichs verwendeten Radius R_d um \mathbf{M}_C , wird eine Bedingung an die Kopfkandidaten gestellt. Befindet sich der Schwerpunkt \mathbf{M}_N innerhalb eines Radius R_d von \mathbf{M}_C entfernt, so ist davon auszugehen, dass der Vektor \mathbf{V}_{CN} keine zuverlässige Orientierung angibt. Entsprechend der Abbildungen 3.33a ist der Bereich der Nüstern nicht hauptsächlich in einem zusammenhängenden Bereich konzentriert, so dass der Vektor \mathbf{V}_{CN} relativ zur Größe der Maske zu kurz ist. Geprüft wird entsprechend $\|\mathbf{V}_{CN}\| = \|\mathbf{M}_N - \mathbf{M}_C\| < R_d$. Sollte die Bedingung erfüllt sein, wird davon ausgegangen, dass es sich nicht um einen Pferdekopf handelt. Die Abbildung 3.33b zeigt den Masseschwerpunkt \mathbf{M}_N (roter Punkt), deren Abstand zu \mathbf{M}_C (blauer Punkt) innerhalb von R_d liegt. Die zuvor noch in Abbildung 3.30a als Kopfkandidaten geführte Region (orange) wird entsprechend als Kopf disqualifiziert.

3. Material und Methoden

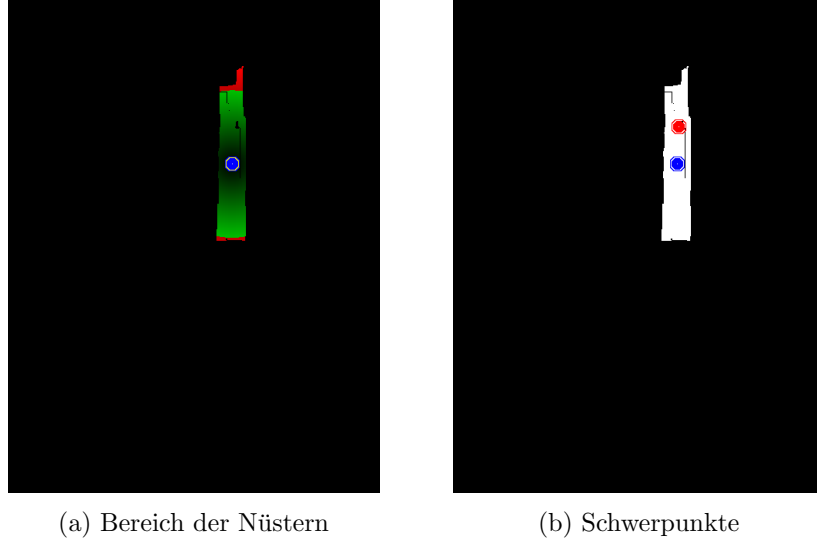


Abbildung 3.33.: Abstand der Masseschwerpunkte $\|\mathbf{M}_N - \mathbf{M}_C\| < R_d$

Ist die genannten Bedingung nicht erfüllt und handelt es sich somit weiterhin um einen Kopfkandidaten, wird der Teil der Maske, welcher entgegen der Richtung von \mathbf{V}_{CN} ausgehend von \mathbf{M}_C liegt, beschränkt (im Normalfall der obere Teil der Maske). Dies dient dazu die Ohren, sollten diese Teil der Maske sein, von der Maske auszuschließen. Die Ohren können sich unabhängig vom Kopf bewegen, welches die Bestimmung der Pose des Kopfes beeinträchtigen kann, solange die Maske die Ohren mit einschließt. Dazu wird ein weiterer prozentualer Schwellwert T_O festgelegt, welcher den Abstand $d_O = T_O \|\mathbf{V}_{CN}\|$ in Abhängigkeit der Länge des Orientierungsvektors \mathbf{V}_{CN} festlegt. Ausgehend vom Punkt \mathbf{M}_C wird die Projektion P_V jedes Pixels der Maske $\mathbf{M}_i = [X_i \ Y_i]^T$ auf den Vektor \mathbf{V}_{CN} durch

$$P_V = \frac{\mathbf{V}_{CN}^T}{\|\mathbf{V}_{CN}\|} (\mathbf{M}_i - \mathbf{M}_C) \quad (3.58)$$

berechnet. Solche Pixel, deren Projektion unterhalb von $P_V < -d_O$ liegen, werden von der Maske ausgeschlossen.

Um die Orientierung des Pferdekopfes im \mathbb{R}^3 zu bestimmen, wird zunächst das arithmetische Mittel aller innerhalb der jeweiligen Maskierung liegenden Normalenvektoren $\mathbf{n}_i = [X_{i,n} \ Y_{i,n} \ Z_{i,n}]^T$, die bereits mit den Ebenenparametern zur Detektion nach Kapitel 3.4.1 bestimmt wurden, berechnet. Ähnlich der Gleichung 3.56 zur Bestimmung des Masseschwerpunkts der Maske im \mathbb{R}^2 , ergibt sich das arithmetische

Mittel der Normalenvektoren $\mathbf{n}_\emptyset = [X_\emptyset \ Y_\emptyset \ Z_\emptyset]^T$ im \mathbb{R}^3 nach

$$X_\emptyset = \frac{\sum_{i=1}^M M_i X_{i,n}}{\sum_{i=1}^M M_i}, \quad Y_\emptyset = \frac{\sum_{i=1}^M M_i Y_{i,n}}{\sum_{i=1}^M M_i} \quad \text{und} \quad Z_\emptyset = \frac{\sum_{i=1}^M M_i Z_{i,n}}{\sum_{i=1}^M M_i}. \quad (3.59)$$

Der so bestimmte Normalenvektor \mathbf{n}_\emptyset dient als erste Schätzung der Orientierung des Pferdekopfes in Richtung der Z-Achse $\mathbf{r}_Z = \mathbf{n}_\emptyset$ im \mathbb{R}^3 . Anschließend werden die Punkte \mathbf{M}_C und \mathbf{M}_N der Maske vom \mathbb{R}^2 in den \mathbb{R}^3 transformiert. Hierzu werden die 3D Koordinaten der Punkte \mathbf{M}_C und \mathbf{M}_N nach Anhang C unter Berücksichtigung der Disparitätenwerte, welche den geringsten Pixelabstand zu den Punkten \mathbf{M}_C und \mathbf{M}_N auf dem Disparitätenbild aufweisen, bestimmt. Die Punkte im \mathbb{R}^3 werden mit \mathbf{X}_C und \mathbf{X}_N bezeichnet. \mathbf{X}_C wird als Position des Kopfes im \mathbb{R}^3 angenommen. Die Orientierung des Kopfes entlang der X-Achse \mathbf{r}_X im \mathbb{R}^3 ist durch den Vektor $\mathbf{v}_{CN} = \mathbf{X}_N - \mathbf{X}_C$ gegeben ($\mathbf{r}_X = \mathbf{v}_{CN} \in \mathbb{R}^3$). Mit dem Gram-Schmidtschen Orthogonalisierungsverfahren [Bronštejn u. a. 2012, S. 323] werden die Anteile des Vektors \mathbf{r}_Z in Richtung des Vektors \mathbf{r}_X entfernt

$$\hat{\mathbf{r}}_Z = \mathbf{r}_Z - \frac{\mathbf{r}_X^T \mathbf{r}_Z}{\mathbf{r}_X^T \mathbf{r}_X} \mathbf{r}_X, \quad (3.60)$$

so dass diese orthogonal zueinander sind ($\hat{\mathbf{r}}_Z^T \mathbf{r}_X = 0$).

Die beiden bisherigen Orientierungsvektoren werden auf die Länge 1 normalisiert:

$$\tilde{\mathbf{r}}_X = \frac{\mathbf{r}_X}{\|\mathbf{r}_X\|} = \frac{\mathbf{r}_X}{\sqrt{\mathbf{r}_X^T \mathbf{r}_X}} \quad \text{und} \quad \tilde{\mathbf{r}}_Z = \frac{\hat{\mathbf{r}}_Z}{\|\hat{\mathbf{r}}_Z\|} = \frac{\hat{\mathbf{r}}_Z}{\sqrt{\hat{\mathbf{r}}_Z^T \hat{\mathbf{r}}_Z}}. \quad (3.61)$$

Der verbleibenden Orientierungsvektor \mathbf{r}_Y der Y-Achse wird durch das Kreuzprodukt (Vektorprodukt [Bronštejn u. a. 2012, S. 188]) aus $\tilde{\mathbf{r}}_X$ und $\tilde{\mathbf{r}}_Z$ berechnet:

$$\mathbf{r}_Y = \tilde{\mathbf{r}}_X \times \tilde{\mathbf{r}}_Z. \quad (3.62)$$

\mathbf{r}_Y besitzt dabei bereits die Länge 1 ($\tilde{\mathbf{r}}_Y = \mathbf{r}_Y$), da die Vektoren $\tilde{\mathbf{r}}_X$ als auch $\tilde{\mathbf{r}}_Z$ die Länge 1 haben und zudem senkrecht aufeinander stehen. Die Transformationsmatrix \mathbf{R}_I , welche die grobe Orientierung des jeweiligen Kopfes komplett beschreibt, ist die Zusammenstellung der einzelnen Orientierungsvektoren $\mathbf{R}_I = [\tilde{\mathbf{r}}_X \ \tilde{\mathbf{r}}_Y \ \tilde{\mathbf{r}}_Z]^T$ in den Spalten der Matrix \mathbf{R}_I . Die Position wird durch die Translation um den Vektor $\mathbf{t}_I = \mathbf{X}_C$ festgelegt.

3.5.2. Posenverfeinerung und Landmarkendetektion

3D Daten zur Posenverfeinerung

Die im vorherigen Kapitel 3.5.1 beschriebene Maske, welche die durch die Detektion bestimmten und vereinzelt Kopfkandidaten markiert, wird zunächst dilatiert. Die Dilatation ist in Gonzalez u. Woods [2008] beschrieben und ist eine morphologische Operation, welche mittels eines Strukturelements den markierten Bereich (Wert 1 in einer Maske) in einem Binärbild (Maske) ausdehnt. Mathematisch wird die Dilatation mit dem Symbol \oplus dargestellt. Ist \mathbf{I} das Binärbild (Maske) und \mathbf{S} das Strukturelement wird die Dilatation mathematisch durch $\mathbf{I} \oplus \mathbf{S}$ beschrieben. Angewendet auf ein Bild wird das Strukturelement zunächst um ein vordefiniertes Zentrumspixel punktgespiegelt, anschließend wird mit dem punktgespiegelten Strukturelement das Binärbild durchlaufen. Beim Durchlaufen werden an der aktuellen Position des Zentrumspixels solche Pixel im dilatierten Bild markiert, bei der mindestens ein Pixel des punktgespiegelten Strukturelements ein markiertes Pixel des ursprünglichen Binärbilds berührt. Angewendet auf ein Grauwertbild entspricht die Dilatation einem Maximum-Filter. Beim Durchlaufen mit dem Strukturelement wird der Wert an der Position des Zentrumspixels mit dem maximalen Wert der Elemente des punktgespiegelten Strukturelements aus dem ursprünglichen Grauwertbild ersetzt.

Die Dilatation der Maske \mathbf{M} aus Kapitel 3.5.1 wird mit einem quadratischen Strukturelement \mathbf{S} durchgeführt. Die Breite des Strukturelements \mathbf{S} entspricht dabei der durchschnittlichen Nachbarschaft, aus den Nachbarschaften, die zur Bestimmung der Ebenenparameter aus Kapitel 3.4.1 innerhalb der Maske angewendet wurden. Die Dilatation $\mathbf{M} \oplus \mathbf{S}$ weitet die Maske auf solche Bereiche des Kopfes aus, welche speziell zur Segmentierung aus Kapitel 3.4.3 verwendet wurden und den abfallenden Bereichen in den Profilen aus Abbildung 3.24 entsprechen.

Abbildung 3.34a zeigt eine Maske \mathbf{M} in weiß, mit dem dilatierten Bereich in rot (rot \cup weiß $\equiv \mathbf{M} \oplus \mathbf{S}$). Zudem ist die Maske in Grün mit dem dilatierten Bereich in Rot über die Tiefeninformationen aus dem Disparitätenbild transformiert auf das Grauwertbild der Farbkamera in Abbildung 3.34b dargestellt. Die 3D Punkte \mathbf{X}_K des jeweiligen Kopfkandidaten werden nach Anhang C mit den Werten des Disparitätenbilds, die auf der dilatierten Maske $\mathbf{M} \oplus \mathbf{S}$ markiert sind, bestimmt.

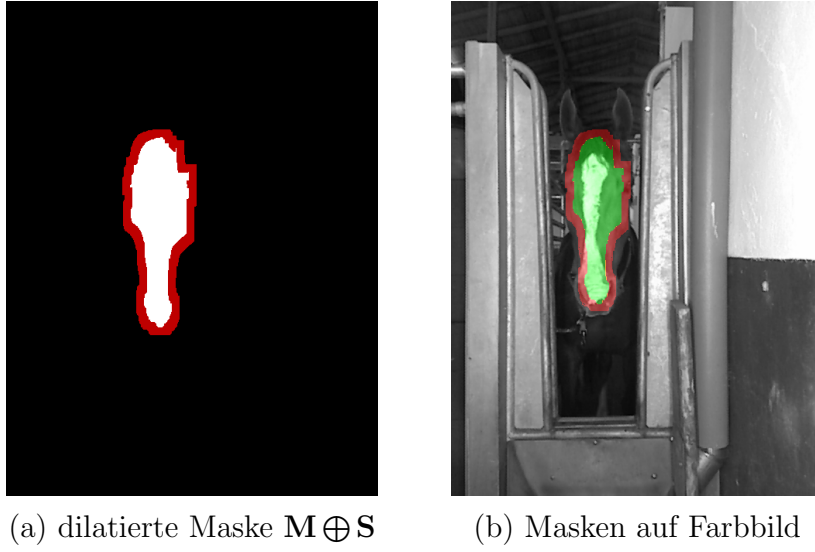


Abbildung 3.34.: Dilatation der einzelnen Maske

Die Abbildungen 3.35a und 3.36a zeigen die 3D Daten der aus dem Disparitätenbild, entsprechend der Maske $\mathbf{M} \oplus \mathbf{S}$ nach Abbildung 3.34a, extrahierten 3D Punkte \mathbf{X}_K der Pferdeköpfe. Die 3D Daten sind als Projektion auf die Kameraebene der Tiefenkamera farbkodiert dargestellt. Abbildungen 3.35b bzw. 3.36b zeigen die synchron aufgenommenen Bilder der Farbkamera.

In den Abbildungen 3.35c und 3.36c sind die 3D Daten \mathbf{X}_{KI} nach der Korrektur $\mathbf{X}_{KI} = \mathbf{R}_I^T (\mathbf{X}_K - \mathbf{t}_I)$ mit den groben Posen aus Kapitel 3.5.1 ebenfalls als Projektion jedoch in einem Abstand von 1 m ($\mathbf{X}_{KI} + [0 \ 0 \ 1]^T$) zur Kameraebene dargestellt.

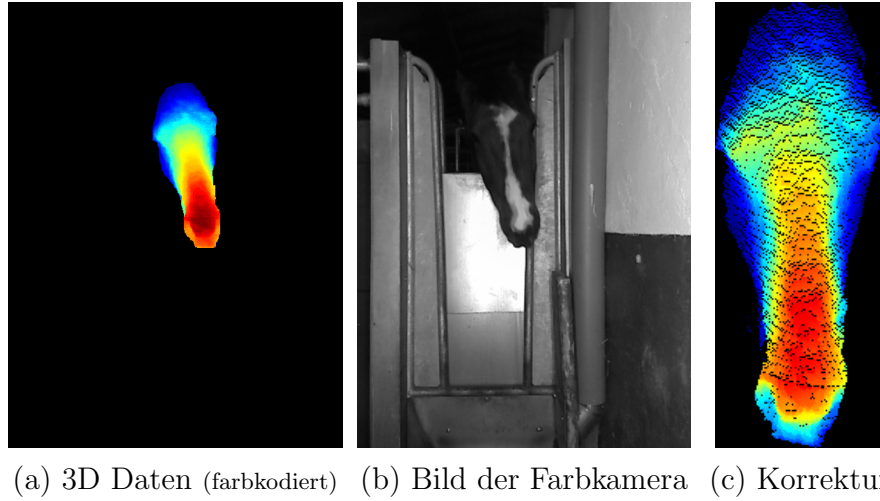


Abbildung 3.35.: 3D Beispieldaten korrigiert mit grober Pose #1

3. Material und Methoden

Die Farbkodierung in den Abbildungen 3.35c und 3.36c entspricht der prozentualen Tiefe aus den Abbildungen 3.35a bzw. 3.36a.

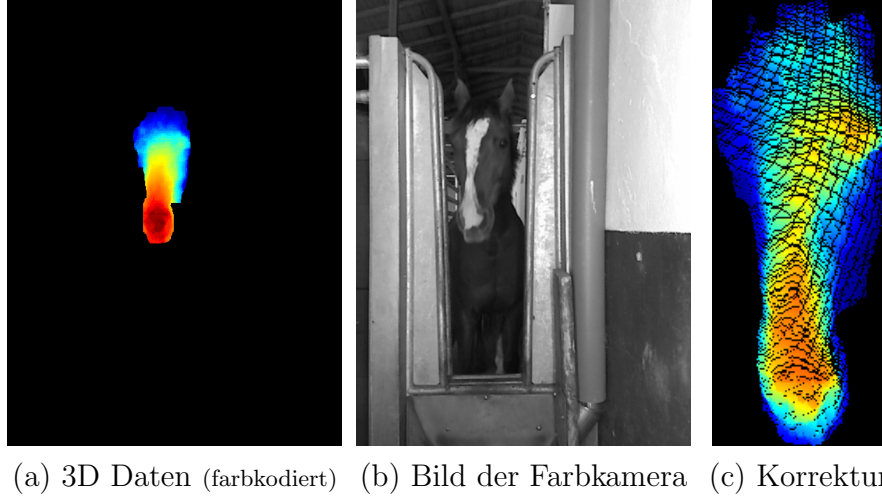


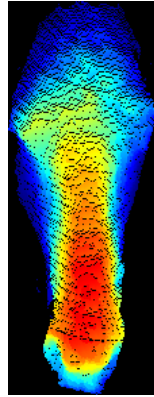
Abbildung 3.36.: 3D Beispieldaten korrigiert mit grober Pose #2

Posenverfeinerung

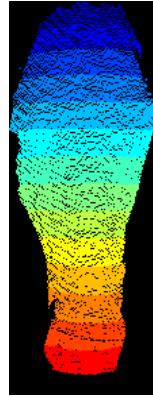
Die grob ausgerichteten 3D Punkte \mathbf{X}_{KI} der Kopfkandidaten werden anschließend durch transverse Schnitte in einzelne Schichten des Kopfes geteilt. Hierzu wird entlang der Längsachse des Kopfes (vertikale Richtung in Abbildung 3.35c bzw. 3.36c) in Teilschritten δ_S verfahren, um alle Punkte \mathbf{X}_{KI} der jeweiligen Schicht $\mathcal{S} := \{\mathbf{X}_{KI} \in \mathcal{S} \mid \|[1 \ 0 \ 0] \mathbf{X}_{KI} - \delta_S\| < \Delta_S/2\}$ zuzuordnen. Die Abbildungen 3.37 und 3.38 zeigen die Unterteilung der 3D Punkte \mathbf{X}_{KI} in einzelne Schichten \mathcal{S} . Die Abbildungen 3.37a und 3.38a entsprechen den korrigierten 3D Daten aus Abbildung 3.35c bzw. 3.35c, in denen die Tiefeninformation farbkodiert dargestellt ist.

Die Abbildungen 3.37b und 3.38b zeigen hingegen die einzelnen Schichten in unterschiedlichen Farben. Die Schichten bestehen in den Abbildungen 3.37b und 3.38b lediglich zur besseren Darstellung aus disjunkten Punktmengen.

Anschließend wird für die Punkte $\mathbf{X}_{KI} \in \mathcal{S}$ mit der Methode von Kasa [1976] (auch bezeichnet als *Kasa – Fit*) ein Kreis \mathbf{K}_S in jeder Schicht bestimmt. Die Kreisparameter sind das Zentrum $\mathbf{K}_C = [Y_{K,C} \ Z_{K,C}]^T$ und der Radius K_R , welche in \mathbf{K}_S zusammengefasst sind. Dies ist in Abbildung 3.39 beispielhaft an drei Schichten dargestellt. Die linke Seite zeigt die einzelnen Schichten, von denen drei als A , B und C markiert sind. Auf der rechten Seite der Abbildung 3.39 sind die zu den Schichten A , B und C gehörenden Kreise mit den Parametern \mathbf{K}_C und K_R dargestellt. Die Kreisparameter \mathbf{K}_S werden algebraisch durch ein lineares Ausgleichsproblem über die Y-Z Koordinaten der Punkte $\mathbf{X}_{KI} \in \mathcal{S}$ bestimmt. Die gewählte Methode von

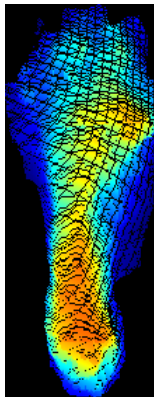


(a) Korrigierte 3D Daten

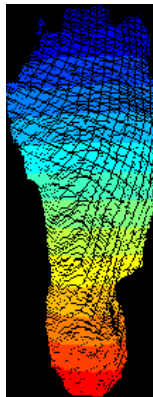


(b) transverse Schnitte

Abbildung 3.37.: Schichten der Beispieldaten #1



(a) Korrigierte 3D Daten



(b) transverse Schnitte

Abbildung 3.38.: Schichten der Beispieldaten #2

3. Material und Methoden

Kasa [1976] ist eine der schnellsten, sie ist jedoch im Vergleich zu den geometrischen Methoden (iterativ) nicht so genau in der Bestimmung der Kreisparameter. So tendiert sie, vorrangig mit Punkten innerhalb unvollständiger Kreise, zu Kreisen mit kleinerem Radius [Al-Sharadqah u. Chernov 2009; Chernov u. Lesort 2005]. In Kapitel 5.4 wird die Auswahl der Methode näher diskutiert und begründet.

Die Kreise \mathbf{K}_S werden mit dem Zentrum der jeweiligen Schicht δ_S um die Position $X_{K,C} = \delta_S$ erweitert, so dass sich die Kreiszentren $\mathbf{K}'_C = [X_{K,C} \ Y_{K,C} \ Z_{K,C}]^T$ im \mathbb{R}^3 befinden. Abbildung 3.40 zeigt in drei Ansichten (a Front-/b Seiten-Ansicht/c Draufsicht) die Kreiszentren \mathbf{K}'_C , welche als rote und grüne Punkte im \mathbb{R}^3 neben den farbkodierten Punktdaten \mathbf{X}_{KI} eines Beispieldatensatzes eingezeichnet sind. Der Kreis \mathbf{K}_S mit dem größten Radius K_R befindet sich in Höhe der Augen (Position $X \approx -0.1$ in Abbildung 3.40) und wird im Folgenden als \mathbf{K}_{max} bezeichnet. Dies ist (nicht nur im Beispieldatensatz) durch die Kopfform der Pferde bestimmt, da die Punktdaten zwischen den Augen im Verhältnis zum restlichen Kopf recht plan sind.

Alle Kreise $\mathbf{K}_{S,N} := \{\mathbf{K}_S \mid X_{K,C} \in \mathbf{K}_S < X'_{K,C} \in \mathbf{K}_{max}\}$, die nach der Koordinate $X_{K,C}$ oberhalb (kleinere Werte $X_{K,C}$) von \mathbf{K}_{max} liegen (rote Punkte in Abbildung 3.40), werden zur weiteren Auswertung nicht mehr verwendet. In die Kreiszentren $\mathbf{K}'_C = [X_{K,C} \ Y_{K,C} \ Z_{K,C}]^T$ der verbleibenden Kreise $\mathbf{K}_{S,P} := \{\mathbf{K}_S \mid X_{K,C} \in \mathbf{K}_S \geq X'_{K,C} \in \mathbf{K}_{max}\}$ wird eine Ebene Π_S eingepasst. Diese wird über ein Ausgleichsproblem entsprechend der Gleichung 3.22 aus Kapitel 3.4.1 bestimmt, so dass sich der Normalenvektor \mathbf{n}_S und der Abstand zum Ursprung δ_S als Parameter der Ebene Π_S ergeben. Die aus den Kreiszentren $\mathbf{K}_{S,P}$ erzeugte Ebene Π_S stellt die Medianebene (sagittale Ebene) des Pferdekopfes (blaues Dreieck in Abbildung 3.40) dar, welche den Pferdekopf entlang der Y-Achse in zwei Teile unterteilt.

In einem nächsten Schritt wird der Masseschwerpunkt \mathbf{t}_A (nach Gleichung 3.59) im \mathbb{R}^3 der Punkte $\mathbf{X}_{KI} = [X_{KI} \ Y_{KI} \ Z_{KI}]^T$ einer Schicht in Höhe der Augen, für die $X_{KI} \geq X'_{K,C}$ & $X_{KI} \leq X'_{K,C} + \Delta_S$ und $(Y_{KI} - Y'_{K,C})^2 + (Z_{KI} - Z'_{K,C})^2 > (K'_R - 0.005\text{ m})^2$ gilt, bestimmt. Der Masseschwerpunkt \mathbf{t}_A stellt zugleich die Transformation zur verfeinerten Pose $\mathbf{t}_{II} = \mathbf{t}_A$ dar und wird bei Anwendung der Korrektur zum neuen Ursprung des Koordinatensystems (Korrektur wird noch nicht angewendet).

Die Medianebene Π_S wird zu $\hat{\Pi}_S$ verfeinert, indem sie parallel in den Punkt \mathbf{t}_{II} verschoben wird, hierzu wird der Abstand $\hat{\delta}_S = \mathbf{n}_S \mathbf{t}_{II}$ angepasst. Der Normalenvektor $\hat{\mathbf{n}}_S$ der neuen Ebene $\hat{\Pi}_S$ entspricht der vorherigen Medianebene $\hat{\mathbf{n}}_S = \mathbf{n}_S$.

Im Anschluss wird eine Auswahl von Punkten auf dem Nasenrücken des Pferdekopfes unter Verwendung der verfeinerten Medianebene $\hat{\Pi}_S$ getroffen. Die Auswahl der Punkte \mathbf{X}_{KI} auf dem Nasenrücken soll einem Streifen mit einer Breite Δ_W und einer Länge Δ_L , welche metrisch angegeben werden, entsprechen. Die Punkte \mathbf{X}_{KI}

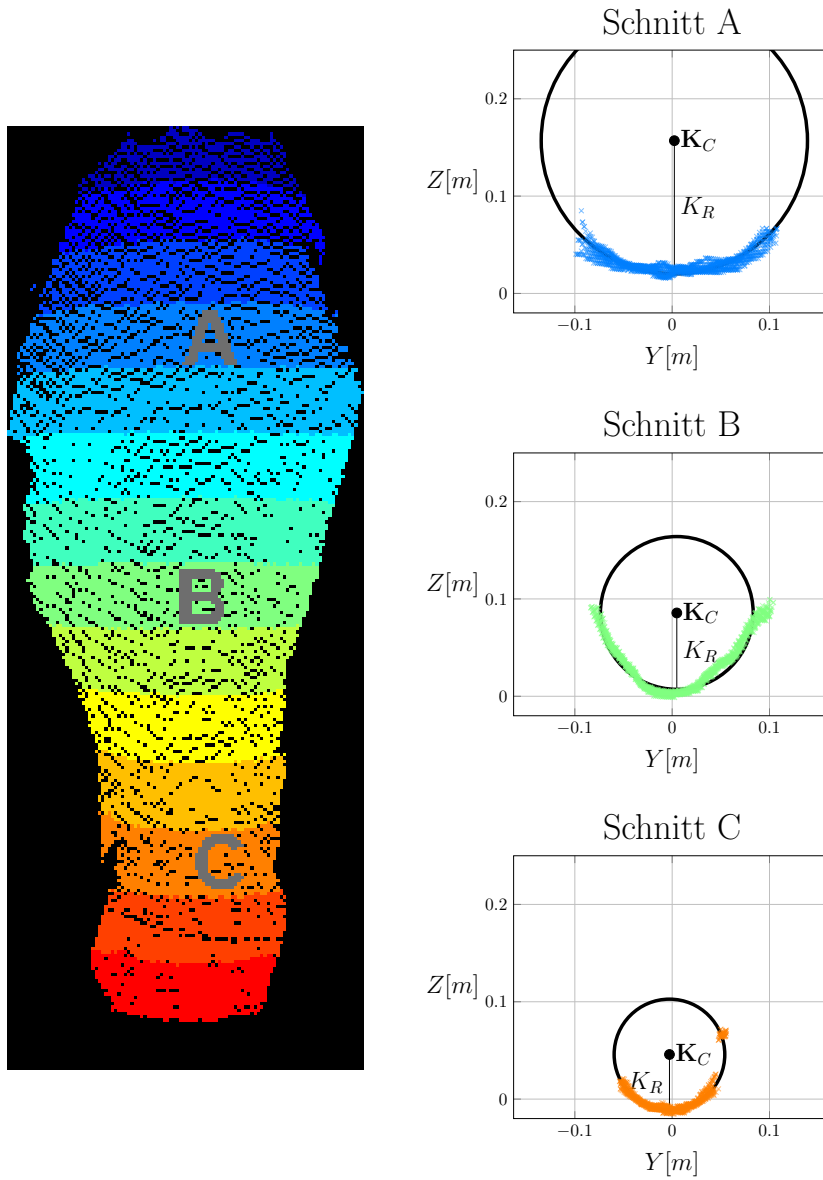


Abbildung 3.39.: Kreise eingepasst in einzelne Schichten (Datensatz #1)

3. Material und Methoden

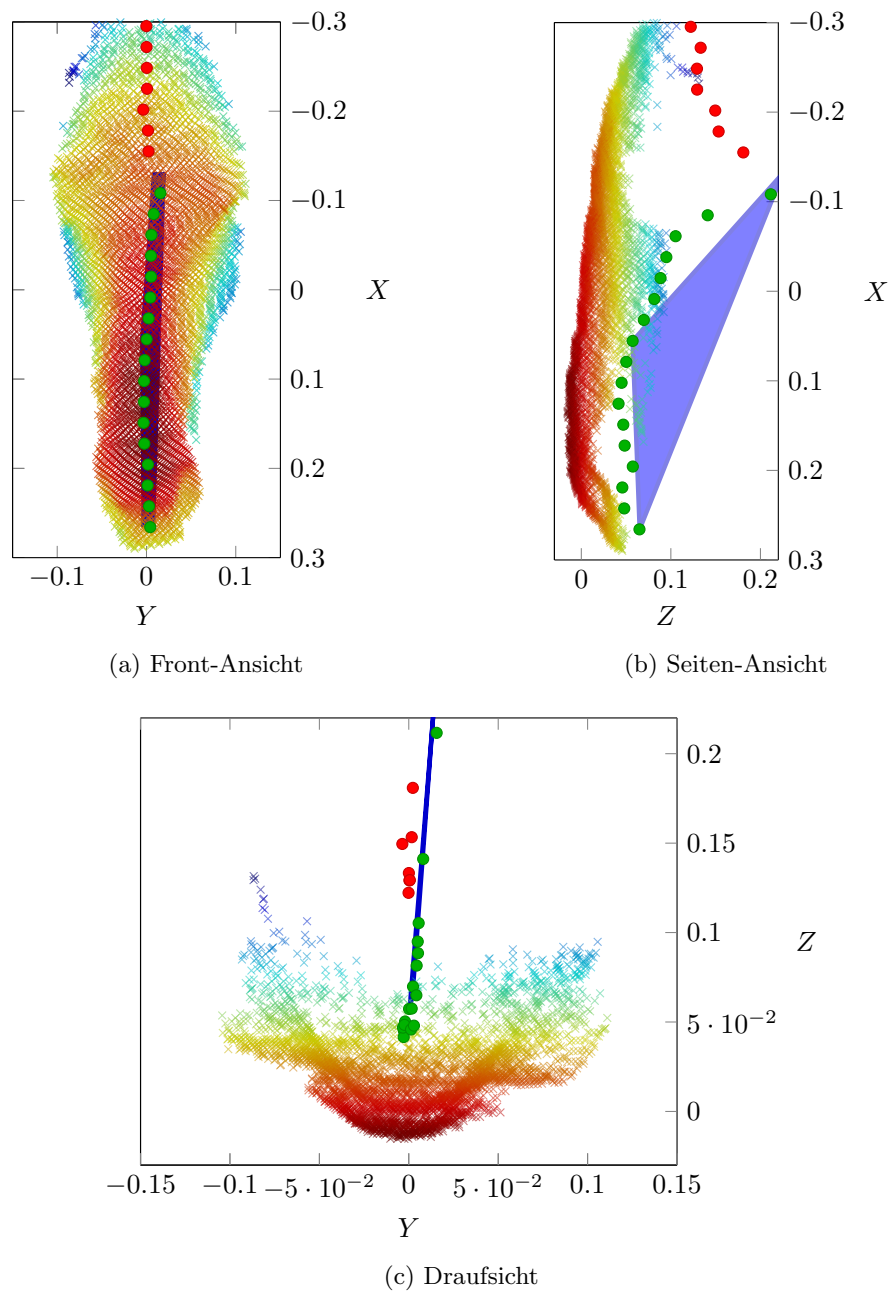


Abbildung 3.40.: Kreiszentren mit Medianebene (Datensatz #1)

in dem Streifen sollen symmetrisch zur Ebene $\hat{\Pi}_S$ sein:

$$\|\mathbf{n}_S^T \mathbf{X}_{KI} - \hat{\delta}_S\| < \frac{\Delta_W}{2} . \quad (3.63)$$

Der Streifen soll zwischen Augen und Nüstern liegen und keine Punkte oberhalb der Augen enthalten ($X'_{K,C}$ entspricht X-Koordinate des Kreiscentrums von \mathbf{K}_{max}):

$$X_{KI} \geq X'_{K,C} . \quad (3.64)$$

Die Länge des Streifens wird durch

$$\|(\mathbf{X}_{KI} - \mathbf{t}_{II})^2\| \leq \Delta_L \quad (3.65)$$

bestränkt. Die Kombination der Ungleichungen 3.63, 3.64 und 3.65 ergibt die Punktauswahl $\mathbf{X}_N \in \mathbf{X}_{KI}$, in die ebenfalls eine Ebene Π_N nach Gleichung 3.22 aus Kapitel 3.4.1 eingepasst wird.

Die Berechnung der Translation \mathbf{t}_{II} wurde bereits beschrieben. Entsprechend dem Kapitel 3.5.1 wird zudem die Rotationsmatrix \mathbf{R}_{II} aus den Normalenvektoren $\hat{\mathbf{n}}_S$ und \mathbf{n}_N der Ebenen $\hat{\Pi}_S$ bzw. Π_N bestimmt.

Die verfeinerte Orientierung des Pferdekopfes im \mathbb{R}^3 entlang der Y-Achse ist durch den Normalenvektor $\mathbf{r}_{YII} = \hat{\mathbf{n}}_S (= \mathbf{n}_S)$ der Medianebene $\hat{\Pi}_S$ (bzw. Π_S) bestimmt. Entlang der Z-Achse ist die Orientierung durch den Normalenvektor $\mathbf{r}_{ZII} = \mathbf{n}_N$ der Ebene Π_N , welche auf die Punkte \mathbf{X}_N eines Streifens auf dem Nasenrücken des Pferdekopfes eingepasst ist, bestimmt. Entsprechend der Gleichung 3.60 aus Kapitel 3.5.1 wird im Anschluss das Gram-Schmidt Orthogonalisierungsverfahren [Bronštejn u. a. 2012, S. 323] eingesetzt, um die zu \mathbf{r}_{YII} orthogonalen Anteile von \mathbf{r}_{ZII} zu extrahieren:

$$\hat{\mathbf{r}}_{ZII} = \mathbf{r}_{ZII} - \frac{\mathbf{r}_{YII}^T \mathbf{r}_{ZII}}{\mathbf{r}_{YII}^T \mathbf{r}_{YII}} \mathbf{r}_{YII} . \quad (3.66)$$

\mathbf{r}_{YII} und $\hat{\mathbf{r}}_{ZII}$ werden entsprechend der Gleichung 3.61 aus Kapitel 3.5.1 auf die Länge 1 gebracht:

$$\tilde{\mathbf{r}}_{YII} = \frac{\mathbf{r}_{YII}}{\|\mathbf{r}_{YII}\|} = \frac{\mathbf{r}_{YII}}{\sqrt{\mathbf{r}_{YII}^T \mathbf{r}_{YII}}} \text{ und } \tilde{\mathbf{r}}_{ZII} = \frac{\hat{\mathbf{r}}_{ZII}}{\|\hat{\mathbf{r}}_{ZII}\|} = \frac{\hat{\mathbf{r}}_{ZII}}{\sqrt{\hat{\mathbf{r}}_{ZII}^T \hat{\mathbf{r}}_{ZII}}} \quad (3.67)$$

Der verbleibenden Orientierungsvektor \mathbf{r}_{XII} der X-Achse wird nach der Gleichung 3.62 aus Kapitel 3.5.1 durch das Kreuzprodukt (Vektorprodukt [Bronštejn u. a. 2012, S. 188]) aus $\tilde{\mathbf{r}}_{YII}$ und $\tilde{\mathbf{r}}_{ZII}$ berechnet:

$$\mathbf{r}_{XII} = \tilde{\mathbf{r}}_{YII} \times \tilde{\mathbf{r}}_{ZII} . \quad (3.68)$$

Die komplette Rotationsmatrix \mathbf{R}_{II} setzt sich wie im vorherigen Kapitel 3.5.1 aus

3. Material und Methoden

den einzelnen Orientierungsvektoren zusammen:

$$\mathbf{R}_{II} = [\mathbf{r}_{XII} \quad \tilde{\mathbf{r}}_{YII} \quad \tilde{\mathbf{r}}_{ZII}] . \quad (3.69)$$

Mit \mathbf{R}_{II} und \mathbf{t}_{II} ist die Beschreibung der verfeinerten Pose nach einer Korrektur mit der groben Pose (aus Kapitel 3.5.1) komplett.

Die Kombination der groben Pose aus Kapitel 3.5.1 (\mathbf{R}_I und \mathbf{t}_I) mit der verfeinerten Pose (\mathbf{R}_{II} und \mathbf{t}_{II}) wird durch

$$\begin{aligned} \mathbf{t}_{I+II} &= \mathbf{t}_I + \mathbf{R}_I \mathbf{t}_{II} \\ \mathbf{R}_{I+II} &= \mathbf{R}_I \mathbf{R}_{II} \end{aligned} \quad (3.70)$$

berechnet.

Landmarkendetektion

Pferdeköpfe besitzen keine der Nase beim Menschen entsprechenden Merkmale, die in den Tiefendaten herausstechen. Das in diesem Unterkapitel beschriebene Vorgehen beschreibt trotzdem eine Möglichkeit, wiederauffindbare Punkte in den 3D Daten eines Pferdekopfes zu extrahieren. Diese Punkte werden als Landmarken bezeichnet und befinden sich in dem hier beschriebenen Fall knapp oberhalb der Augen. Die Extraktion dieser Landmarken bedarf der in dem vorherigen Unterkapitel *Posenverfeinerung* beschriebenen Posenkorrektur. Die Korrektur ist bereits darauf ausgelegt grob den Mittelpunkt zwischen beiden Augen in den Ursprung des korrigierten Koordinatensystems zu legen. Die Position der Augen wurden dazu näherungsweise über den Kreis, welcher den maximalen Radius beim Einpassen in die einzelnen Schichten (entsprechend der Abbildung 3.39) aufwies, bestimmt.

Zum Auffinden der Landmarken werden die nach dem Unterkapitel *Posenverfeinerung* korrigierten 3D Daten \mathbf{X}_{KII} verwendet, deren Tiefeninformation in Abbildung 3.41 mit unterschiedlichen Grautönen kodiert dargestellt ist. Die korrigierten Daten \mathbf{X}_{KII} werden entsprechend aus den grob korrigierten Daten (Unterkapitel *3D Daten zur Posenverfeinerung*) \mathbf{X}_{KI} durch $\mathbf{X}_{KII} = \mathbf{R}_{II}^T (\mathbf{X}_{KI} - \mathbf{t}_{II})$ oder aus den unkorrigierten Daten \mathbf{X}_K durch $\mathbf{X}_{KII} = \mathbf{R}_{I+II}^T (\mathbf{X}_K - \mathbf{t}_{I+II})$ berechnet.

Aus den korrigierten Daten \mathbf{X}_{KII} wird ein Bereich um die Augen extrahiert, dieser Bereich ist in Abbildung 3.41 farbkodiert dargestellt und gegenüber den restlichen 3D Daten des Pferdekopfes (graue Darstellung) hervorgehoben. Die Daten $\tilde{\mathbf{X}}_A = [\tilde{X}_A \quad \tilde{Y}_A \quad \tilde{Z}_A]^T$ im Bereich der Augen $\tilde{\mathcal{A}}$ werden über die X-Koordinate aus allen \mathbf{X}_{KII} bestimmt $\tilde{\mathcal{A}} := \{\mathbf{X}_{KII} \in \tilde{\mathcal{A}} \mid \|X_{KII}\| < \Delta_A\}$.

In die Punkte $\tilde{\mathbf{X}}_A \in \tilde{\mathcal{A}}$ wird ein Kreis \mathbf{K}_A durch das im Unterkapitel *Posenverfeinerung* beschriebene Ausgleichsproblem der Y-Z Koordinaten eingepasst. Der Kreis

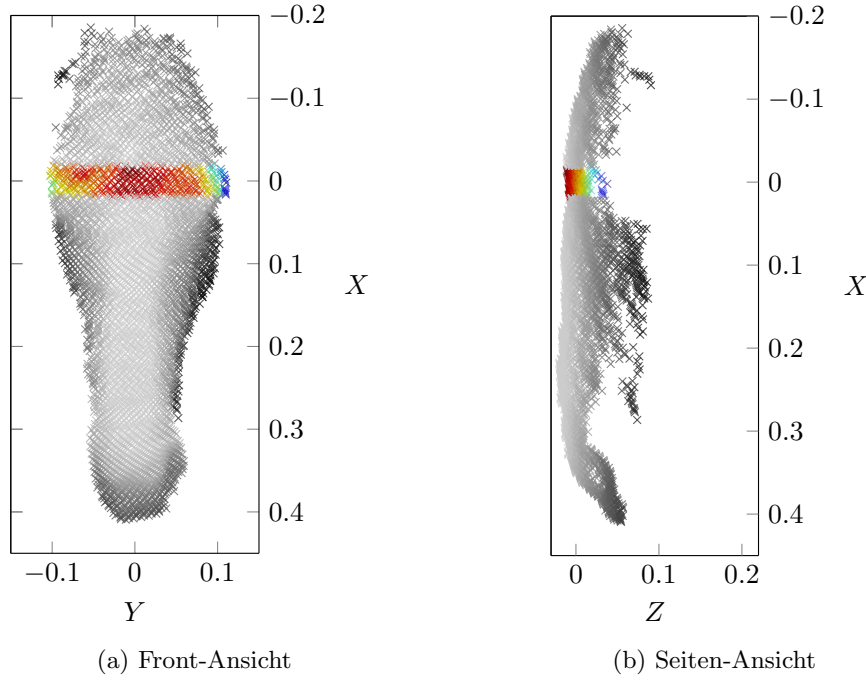


Abbildung 3.41.: Bereich zur Bestimmung der Landmarken (Datensatz #1)

\mathbf{K}_A hat sein Zentrum in $\mathbf{K}_{C,A} = [Y_{A,C} \ Z_{A,C}]^T$ (auf der Y-Z Ebene \mathbb{R}^2) und einen Radius von $K_{R,A}$. Zudem wird eine Linie \mathbf{L}_A ebenfalls in die Y-Z Koordinaten der Punkte $\tilde{\mathbf{X}}_A$ über das Ausgleichproblem $L_M \tilde{Y}_A + L_B = \tilde{Z}_A$ bestimmt. L_M und L_B sind die Parameter der Linie \mathbf{L}_A . L_M ist die Steigung der Linie \mathbf{L}_A , wobei L_B den Schnittpunkt der Z-Achse mit der Linie \mathbf{L}_A beschreibt. In Abbildung 3.42a ist der Kreis \mathbf{K}_A in Blau und die Linie \mathbf{L}_A in Cyan in einer Draufsicht der Punkte $\tilde{\mathbf{X}}_A$ eingezeichnet. Die Punktdaten $\tilde{\mathbf{X}}_A$ sind, wie zuvor, entsprechend der Tiefeninformation (Z-Achse) farbkodiert dargestellt. Die schraffierten Bereiche \mathcal{A} enthalten die Punkte $\mathbf{X}_A \in \mathcal{A}$, welche ausserhalb des Kreises \mathbf{K}_A und oberhalb der Linie \mathbf{L}_A liegen. Die Punkte $\tilde{\mathbf{X}}_A$, welche die Bedingungen

$$\begin{aligned}
 (\tilde{Y}_A - Y_{A,C})^2 + (\tilde{Z}_A - Z_{A,C})^2 &> K_{R,A}^2 \quad \& \\
 L_M \tilde{Y}_A + L_B &> \tilde{Z}_A,
 \end{aligned} \tag{3.71}$$

die sich mit den Parametern von \mathbf{K}_A und \mathbf{L}_A aufstellen lassen, erfüllen, ergeben dabei die Punkte $\mathbf{X}_A \in \mathcal{A}$. Die Punkte \mathbf{X}_A sind in der Abbildung 3.42b am Beispiel des ersten Datensatzes in Grün, die verbleibenden Punkte von $\tilde{\mathbf{X}}_A$ ($\tilde{\mathbf{X}}_A/\mathbf{X}_A$) in Grau dargestellt. Die Skalierung zwischen Y und Z Achse ist in der Draufsicht aus Abbildung 3.42 verändert und ist nicht identisch, wie es in der Abbildung 3.40c des vorherigen Kapitels der Fall ist.

3. Material und Methoden

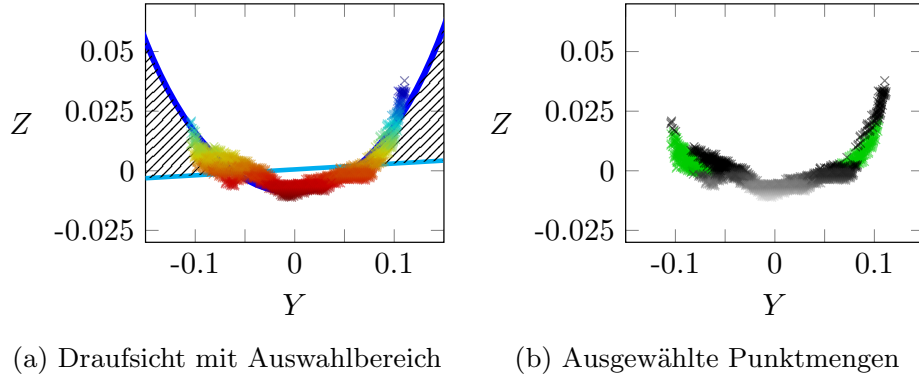


Abbildung 3.42.: Extrahieren der Punktmengen (Datensatz #1)

Über die Y-Koordinate werden die Punktdaten $\mathbf{X}_A = [X_A \ Y_A \ Z_A]^T \in \mathcal{A}$ anschließend in zwei Teilmengen \mathcal{A}_L und \mathcal{A}_R unterteilt. Die Punkte $\mathbf{X}_L \in \mathcal{A}_L$ bilden die Datenbasis zur Bestimmung der linken Landmarke (ausgehend vom Tier), deren Bereich \mathcal{A}_L durch $\mathcal{A}_L := \{\mathbf{X}_A \in \mathcal{A}_L \mid Y_A > 0\}$ bestimmt wird. Entsprechend sind die Punkte $\mathbf{X}_R \in \mathcal{A}_R$ die Datenbasis zur rechten Landmarke (ausgehend vom Tier). Der Bereich \mathcal{A}_R ist durch $\mathcal{A}_R := \{\mathbf{X}_A \in \mathcal{A}_R \mid Y_A < 0\}$ gegeben.

Die Landmarken \mathbf{P}_L und \mathbf{P}_R werden im \mathbb{R}^3 durch den Masseschwerpunkt der Punkte $\mathbf{X}_L \in \mathcal{A}_L$ bzw. $\mathbf{X}_R \in \mathcal{A}_R$ bestimmt. Die Berechnung des Masseschwerpunkts $\mathbf{P}_L = [X_{P,L} \ Y_{P,L} \ Z_{P,L}]^T$ der Punkte $\mathbf{X}_L = [X_L \ Y_L \ Z_L]^T$ im \mathbb{R}^3 ist der Gleichung 3.56 im \mathbb{R}^2 sehr ähnlich und entspricht der Gleichung 3.59 zur Berechnung des arithmetischen Mittels im \mathbb{R}^3 , jedoch ohne eine Maskierung:

$$\begin{aligned}
 X_{P,L} &= \frac{1}{|\mathcal{A}_L|} \sum_{\forall \mathbf{X}_L \in \mathcal{A}_L} X_L, \\
 Y_{P,L} &= \frac{1}{|\mathcal{A}_L|} \sum_{\forall \mathbf{X}_L \in \mathcal{A}_L} Y_L \text{ und} \\
 Z_{P,L} &= \frac{1}{|\mathcal{A}_L|} \sum_{\forall \mathbf{X}_L \in \mathcal{A}_L} Z_L.
 \end{aligned} \tag{3.72}$$

Dabei ist $|\mathcal{A}_L|$ die Anzahl der Punkte \mathbf{X}_L in \mathcal{A}_L . Entsprechend der Gleichung 3.72 berechnet sich $\mathbf{P}_R = [X_{P,R} \ Y_{P,R} \ Z_{P,R}]^T$ aus den Punkten $\mathbf{X}_R = [X_R \ Y_R \ Z_R]^T \in$

\mathcal{A}_R durch

$$\begin{aligned} X_{P,R} &= \frac{1}{|\mathcal{A}_R|} \sum_{\forall \mathbf{X}_R \in \mathcal{A}_R} X_R , \\ Y_{P,R} &= \frac{1}{|\mathcal{A}_R|} \sum_{\forall \mathbf{X}_R \in \mathcal{A}_R} Y_R \text{ und} \\ Z_{P,R} &= \frac{1}{|\mathcal{A}_R|} \sum_{\forall \mathbf{X}_R \in \mathcal{A}_R} Z_R . \end{aligned} \tag{3.73}$$

Die Landmarken \mathbf{P}_L und \mathbf{P}_R sind so in den durch $\mathbf{R}_{\mathbf{II}}$ bzw. $\mathbf{t}_{\mathbf{II}}$ korrigierten Daten gegeben.

Finale Posenverfeinerung mit Landmarken

Die Abschließende Posenverfeinerung sieht eine Rotation um die Z-Achse vor, die den Masseschwerpunkt \mathbf{P}_N der Punkte $\mathbf{X}_N \in \mathcal{N}$ auf die X-Achse rotieren. Hierzu wird der Bereich $\mathcal{N} := \{\mathbf{X}_{II} \in \mathcal{N} \mid Z_{KII} < 0 \& X_{KII} > \Delta_L\}$ aus den nach der *Posenverfeinerung* korrigierten Daten \mathbf{X}_{KII} bestimmt. Δ_L entspricht der Länge des Streifens aus Gleichung 3.65, welcher im Unterkapitel *Posenverfeinerung* zur Korrektur verwendet wurde. Der Bereich \mathcal{N} entspricht damit dem Bereich jenseits des zur *Posenverfeinerung* verwendeten Streifens, der in Abbildung 3.43, neben den Punkten \mathbf{X}_{KII} in Grau, farbkodiert dargestellt ist. \mathcal{N} stellt damit einen zentralen Bereich auf dem Nasenrücken dar, der sich oberhalb der Nüstern befindet.

3. Material und Methoden

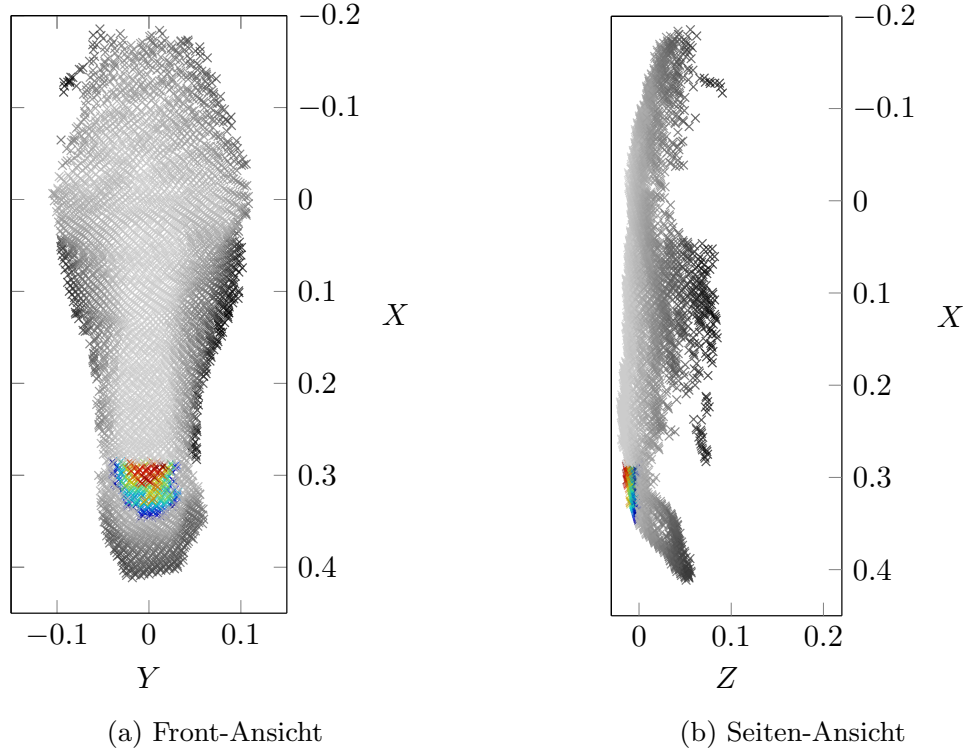


Abbildung 3.43.: Zentraler Bereich \mathcal{N} auf Nasenrücken (Datensatz #1)

Der Masseschwerpunkt $\mathbf{P}_N = [X_{P,N} \ Y_{P,N} \ Z_{P,N}]^T$ wird entsprechend der Gleichung 3.72 durch

$$\begin{aligned} X_{P,N} &= \frac{1}{|\mathcal{N}|} \sum_{\forall \mathbf{X}_N \in \mathcal{N}} X_N, \\ Y_{P,N} &= \frac{1}{|\mathcal{N}|} \sum_{\forall \mathbf{X}_N \in \mathcal{N}} Y_N \text{ und} \\ Z_{P,N} &= \frac{1}{|\mathcal{N}|} \sum_{\forall \mathbf{X}_N \in \mathcal{N}} Z_N \end{aligned} \quad (3.74)$$

bestimmt. Aus dem Winkel zwischen X-Achse und der Verbindung von Ursprung mit dem Punkt \mathbf{P}_N in der X-Y Ebene wird der Rotationswinkel zur Korrektur bestimmt:

$$\alpha_Z = \text{atan}\left(\frac{Y_{P,N}}{X_{P,N}}\right). \quad (3.75)$$

Die Rotationsmatrix zur Korrektur \mathbf{R}_{III} wird zur Rotation um die Z-Achse mit

$$\mathbf{R}_{\text{III}} = \begin{bmatrix} \cos(\alpha_Z) & -\sin(\alpha_Z) & 0 \\ \sin(\alpha_Z) & \cos(\alpha_Z) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.76)$$

aufgestellt.

Die Translation \mathbf{t}_{III} verschiebt den Mittelpunkt der beiden Landmarken aus dem Unterkapitel *Landmarkendetektion* durch

$$\mathbf{t}_{\text{III}} = \frac{\mathbf{P}_L + \mathbf{P}_R}{2} \quad (3.77)$$

zur Korrektur in den Ursprung.

Die Landmarken \mathbf{P}_L und \mathbf{P}_R werden durch

$$\mathbf{P}_{LIII} = \mathbf{R}_{\text{III}}^T (\mathbf{P}_{LIII} - \mathbf{t}_{\text{III}}) \text{ und } \mathbf{P}_{RIII} = \mathbf{R}_{\text{III}}^T (\mathbf{P}_{RIII} - \mathbf{t}_{\text{III}}) \quad (3.78)$$

in den korrigierten Raum transformiert.

Die Translation wird genutzt, um die Augen (bzw. die Landmarken) in eine einheitliche Position im Raum zu verschieben, wobei die Rotation noch bestehende Abweichungen des Nasenrückens zur X-Achse korrigieren soll.

Die Kombination der Korrekturen aus $\mathbf{R}_{\text{I+II}}$, $\mathbf{t}_{\text{I+II}}$, \mathbf{R}_{III} und \mathbf{t}_{III} ergibt sich durch

$$\begin{aligned} \mathbf{t}_{\text{I+II+III}} &= \mathbf{t}_{\text{I+II}} + \mathbf{R}_{\text{I+II}} \mathbf{t}_{\text{III}} \\ \mathbf{R}_{\text{I+II+III}} &= \mathbf{R}_{\text{I+II}} \mathbf{R}_{\text{III}} . \end{aligned} \quad (3.79)$$

Abbildung 3.44 zeigt die Landmarken (\mathbf{P}_{LIII} und \mathbf{P}_{RIII}) der *Landmarkendetektion* (vorheriges Unterkapitel), welche nach der Umkehr der Korrektur mit $\mathbf{R}_{\text{I+II+III}}$ und $\mathbf{t}_{\text{I+II+III}}$ auf die Bilder der Farbkamera projiziert wurden. Die Landmarken selber sind als kleine rote Quadrate dargestellt. Die projizierten Datenbasen \mathcal{A}_L und \mathcal{A}_R (Unterkapitel *Landmarkendetektion*) sind als grüne Bereiche in Abbildung 3.44 gezeigt.

3. Material und Methoden



(a) Landmarken Datensatz #1

(b) Landmarken Datensatz #2

Abbildung 3.44.: Landmarken projiziert auf die Bilder der Farbkamera

3.5.3. Normalisierung der 2D Daten anhand der 3D Informationen

Die Pferdeköpfe werden mit den Verfahren aus Kapitel 3.4 in den Bilddaten aufgefunden, um anschließend deren Pose mit den in den Kapiteln 3.5.1 und 3.5.2 beschriebenen Methoden zu bestimmen. Die Grauwertbilder der Pferdeköpfe sollen nun durch eine virtuelle Kameraansicht so in eine Frontansicht transformiert werden, als würden die Köpfe direkt in die Kamera blicken.

Virtuelle Kamera

Ein Ansatz wäre die Projektive Transformation im \mathbb{P}^2 , wie sie im Anhang B zu finden ist. Durch die Projektive Transformation wird eine lineare Transformation von Punkten einer Ebene im \mathbb{R}^3 auf eine zweite Ebene (Homographie) beschrieben [Faugeras u. a. 2004; Hartley u. Zisserman 2004]. Dies führt dazu, dass nur Punkte auf einer Ebene, also planare Objekte im \mathbb{R}^3 , sich durch die Projektive Transformation korrekt transformieren lassen. Handelt es sich jedoch um ein dreidimensionales Objekt, wie den Pferdekopf, dessen Pose virtuell korrigiert werden soll, so reicht eine Projektive Transformation im \mathbb{P}^2 nicht aus.

Daher ist es nötig die Tiefeninformationen des Objektes zu berücksichtigen, um eine Korrektur der Grauwertbilder nach der in den Kapiteln 3.5.1 und 3.5.2 bestimmten

Pose durchführen zu können. Aus der Tiefe werden, wie zuvor zur Posendetektion (Kapitel 3.5.2), die 3D Daten \mathbf{X}_K der Pferdeköpfe nach Anhang C bestimmt. Die Pose, welche durch die Rotation $\mathbf{R}_{\mathbf{I}+\mathbf{II}+\mathbf{III}}$ und die Translation $\mathbf{t}_{\mathbf{I}+\mathbf{II}+\mathbf{III}}$ beschrieben ist, lässt sich, wie zuvor im Kapitel 3.5.2 (mit $\mathbf{R}_{\mathbf{I}}$ und $\mathbf{t}_{\mathbf{I}}$), durch $\mathbf{X}_{KIII} = \mathbf{R}_{\mathbf{I}+\mathbf{II}+\mathbf{III}}^T (\mathbf{X}_K - \mathbf{t}_{\mathbf{I}+\mathbf{II}+\mathbf{III}})$ korrigieren. Die virtuelle Kamera wird entsprechend der Pose des Kopfes so positioniert, als würde sie den Pferdekopf frontal aus 1 m Entfernung aufnehmen. Hierzu werden die extrinsischen Kameraparameter durch die Orientierung

$$\mathbf{R}_V = \mathbf{R}_{\mathbf{I}+\mathbf{II}+\mathbf{III}}^T \quad (3.80)$$

und die Translation

$$\mathbf{t}_V = [0 \ 0 \ 1]^T - \mathbf{R}_{\mathbf{I}+\mathbf{II}+\mathbf{III}}^T \mathbf{t}_{\mathbf{I}+\mathbf{II}+\mathbf{III}} \quad (3.81)$$

festgelegt, wodurch die in das Kamerakoordinatensystem nach Gleichung 3.6 (Kapitel 3.3.2) abgebildeten Daten \mathbf{X}_V der virtuellen Kamera

$$\mathbf{X}_V = \mathbf{R}_V \mathbf{X}_K + \mathbf{t}_V \quad (3.82)$$

den korrigierten Daten \mathbf{X}_{KIII} mit einem Versatz um 1 m entsprechen:

$$\mathbf{X}_V = \mathbf{X}_{KIII} + [0 \ 0 \ 1]^T. \quad (3.83)$$

Zur Bestimmung der intrinsischen Parameter der virtuellen Kamera wird die Distanz Δ_P der Landmarken \mathbf{P}_{LIII} und \mathbf{P}_{RIII} , welche in Kapitel 3.5.2 beschrieben sind, auf der Y-Z Ebene bestimmt:

$$\Delta_P = \| [0 \ 1 \ 1] \mathbf{P}_{LIII} - [0 \ 1 \ 1] \mathbf{P}_{RIII} \| . \quad (3.84)$$

Die Kameramatrix \mathbf{K}_V , welche die intrinsischen Parameter der virtuellen Kamera enthält, soll durch die Projektion der Distanz Δ_P einem vorgegebenen Pixelabstand Δ_{px} im normalisierten Grauwertbild entsprechen. Der Parameter α_v wird in der Kameramatrix nach Gleichung 3.5 aus Kapitel 3.3.2 für beide Koordinatenachsen ($\alpha_x = \alpha_y = \alpha_v$) verwendet

$$\mathbf{K}_V = \begin{pmatrix} \alpha_v & s_v & x_v \\ 0 & \alpha_v & y_v \\ 0 & 0 & 1 \end{pmatrix} \quad (3.85)$$

und durch das Verhältnis $\alpha_v = \frac{\Delta_{px}}{\Delta_P}$ bestimmt. Es wird keine Scherung angenommen $s_v = 0$. x_v und y_v entsprechen der Projektion des Mittelpunktes der beiden Landmarken (\mathbf{P}_{LIII} und \mathbf{P}_{RIII}) im normalisierten Grauwertbild.

3. Material und Methoden

Durch die Projektion der 3D Kopfdaten \mathbf{X}_K auf die Kameraebene der virtuellen Kamera entsprechend der Gleichung 3.8 (Kapitel 3.3.2)

$$\mathbf{x}_v = \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \sim \begin{pmatrix} \alpha_v & 0 & x_v \\ 0 & \alpha_v & y_v \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{bmatrix} \mathbf{R}_V & \mathbf{t}_V \\ 0 & 1 \end{bmatrix} \begin{pmatrix} X_K \\ Y_K \\ Z_K \\ 1 \end{pmatrix} \quad (3.86)$$

lassen sich die jeweiligen Pixel im normalisierten Grauwertbild bestimmen, in die die Punkte \mathbf{X}_K projiziert werden. Für das virtuelle Bild wird eine Tiefenkarte generiert, indem in jeden Pixel \mathbf{x}_v die jeweilige Tiefe Z_{KIII} der korrigierten Daten \mathbf{X}_{KIII} eingetragen wird. Entsprechende Tiefenkarten sind bereits im Kapitel 3.5.1 in den Abbildungen 3.35c bzw. 3.36c gezeigt. Auch wenn die Abbildungen lediglich die grobe Posenkorrektur zeigen, sind die Löcher innerhalb des Pferdekopfes erkennbar (schwarze Punkte im ansonsten farbkodierten Tiefenbild). Je größer das Tiefenbild bzw. das im Folgenden zu erzeugende normalisierte Grauwertbild werden soll, umso grösser werden die Löcher, da nur eine begrenzte Anzahl an Tiefendaten \mathbf{X}_K aus dem Disparitätenbild zur Verfügung steht. Bei kleinen, vereinzelt auftretenden Löchern kann zur Ausbesserung des Tiefenbildes ein Medianfilter eingesetzt werden [Rana u. Flierl 2011; Lee u. Ho 2011], welches in diesem Fall nicht ausreicht, da möglichst große Normbilder extrahiert werden sollen.

Ausbessern der virtuellen Tiefenkarte

Die Löcher in der virtuellen Tiefenkarte werden daher durch eine bilineare Interpolation ausgebessert. Ein entsprechendes Verfahren wird in [Lee u. Ho 2009] zur Korrektur der Tiefenkarte eingesetzt. Lee u. Yoo [2014] stellt einen Vergleich zwischen der bilineare Interpolation und weiteren Verfahren zusammen, wobei die bilineare Interpolation vorrangig in den Randbereichen schlechte Ergebnisse liefert. Die Betrachtung der Randbereiche wurde in der für diesen Fall beschriebenen Normalisierung jedoch ignoriert, da lediglich ein Objekt (der Pferdekopf) zu normalisieren ist. Der Pferdekopf selber hat jedoch nur an den Randbereichen zu anderen Objekten starke Tiefenunterschiede, so dass entsprechende Filter- und In-Painting-Methoden zur Ausbesserung von Verdeckungen im virtuellen Bild aus den erwähnten Arbeiten [Rana u. Flierl 2011; Lee u. Ho 2011; Lee u. Yoo 2014] hier unberücksichtigt bleiben. Zudem werden nur Posen berücksichtigt, in denen keine Selbstverdeckung am Pferdekopf auftritt (siehe Kapitel 4.3).

Um die Datenbasis der verfügbaren Tiefendaten \mathbf{X}_V zu vergrößern, wird die Symmetrie der Pferdeköpfe entlang der Y-Achse ausgenutzt. Die 3D Daten \mathbf{X}_V werden dazu an der X-Z-Ebene gespiegelt $\mathbf{X}_{V\star}$, um die ursprünglichen Daten \mathbf{X}_V mit diesen zu ergänzen \mathbf{X}_{V+} . Die Projektion der Punkte $\mathbf{X}_{V\star}$ nach Gleichung 3.86 kann auf Pixelbasis durch $v_\star = 2y_v - v$ berechnet werden ($x_\star = x$).

Die bilineare Interpolation wird auf Basis der nach Gleichung 3.86 projizierten Punkte \mathbf{X}_{V+} , deren Pixel \mathbf{x}_{v+} die Tiefeninformation Z_{V+} in dem Tiefenbild halten, mit dem MATLAB[®]-Befehl *griddata* durchgeführt. Farin [1996]; Amidror [2002]; Shirley u. Marschner [2009] beschreiben die Methode in der Baryzentrische Koordinaten (*barycentric coordinates*) genutzt werden, um die Punkte innerhalb der durch eine Delaunay-Triangulation erzeugten Dreiecke zu interpolieren. Die Delaunay-Triangulation [O'Rourke 1998; Aurenhammer 1991], welche in der Bibliothek QHull verfügbar ist Barber u. a. [1996], wird ebenfalls im Anhang D zur Registrierung mit dem Iterative Closest Point - Algorithmus verwendet. Abbildung 3.45 zeigt beispielhaft ein Dreieck, welches drei durch die Delaunay-Triangulation zusammenhängende Punkte $\mathbf{x}_{v+,i}$ darstellt. \mathbf{x}_i ist der zu interpolierende Punkt, welcher innerhalb des Dreiecks liegt, wodurch er sich durch eine Baryzentrische Koordinate $\overset{\circ}{\mathbf{x}}_i$ beschreiben lässt.

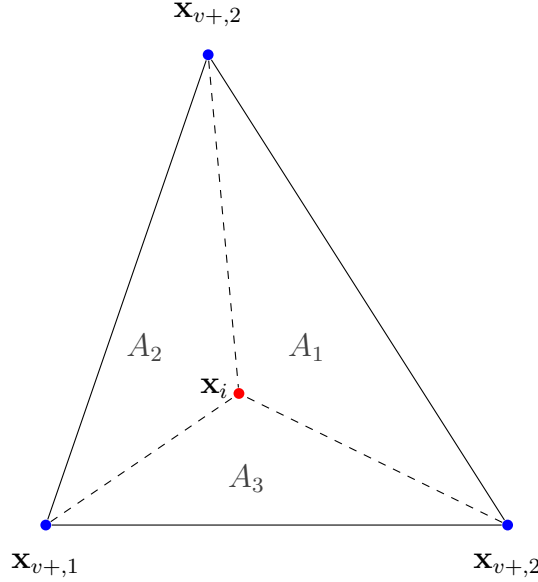


Abbildung 3.45.: Baryzentrische Koordinate innerhalb einer Delaunay-Triangulation
[Verändert nach: Shirley u. Marschner 2009]

Die Barizentrische Koordinate $\overset{\circ}{\mathbf{x}}_i = [\alpha_1 \ \alpha_2 \ \alpha_3]^T$ beschreibt den zu interpolierenden Punkt \mathbf{x}_i durch die lineare Kombination der Punkte $\mathbf{x}_{v+,i}$ des Triangulationsdreiecks

$$\begin{aligned} \mathbf{x}_i &= [\mathbf{x}_{v+,1} \ \mathbf{x}_{v+,2} \ \mathbf{x}_{v+,3}] \overset{\circ}{\mathbf{x}}_i \\ &= \alpha_1 \mathbf{x}_{v+,1} + \alpha_2 \mathbf{x}_{v+,2} + \alpha_3 \mathbf{x}_{v+,3} . \end{aligned} \quad (3.87)$$

3. Material und Methoden

Die Barizentrische Koordinate $\overset{\circ}{\mathbf{x}}_i$ ist die Lösung des linearen Gleichungssystems, welches durch Gleichung 3.88 aufgestellt wird. Mit der Cramerschen Regel [Bronštejn u. a. 2012, S. 318] lässt sich die Lösung und damit die Koordinate $\overset{\circ}{\mathbf{x}}_i$ durch

$$\begin{aligned}\alpha_1 &= \frac{D_1}{D}, \\ \alpha_2 &= \frac{D_2}{D} \text{ und} \\ \alpha_3 &= \frac{D_3}{D}\end{aligned}\tag{3.88}$$

aus den Determinanten

$$\begin{aligned}D &= \begin{vmatrix} x_{v+,1} & x_{v+,2} & x_{v+,3} \\ y_{v+,1} & y_{v+,2} & y_{v+,3} \\ 1 & 1 & 1 \end{vmatrix}, \\ D_1 &= \begin{vmatrix} x_i & x_{v+,2} & x_{v+,3} \\ y_i & y_{v+,2} & y_{v+,3} \\ 1 & 1 & 1 \end{vmatrix}, \\ D_2 &= \begin{vmatrix} x_{v+,1} & x_i & x_{v+,3} \\ y_{v+,1} & y_i & y_{v+,3} \\ 1 & 1 & 1 \end{vmatrix} \text{ und} \\ D_3 &= \begin{vmatrix} x_{v+,1} & x_{v+,2} & x_i \\ y_{v+,1} & y_{v+,2} & y_i \\ 1 & 1 & 1 \end{vmatrix}\end{aligned}\tag{3.89}$$

bestimmen. Dies entspricht geometrisch dem Verhältnis der Flächen $\alpha_1 = \frac{A_1}{A}$, $\alpha_2 = \frac{A_2}{A}$ und $\alpha_3 = \frac{A_3}{A}$ aus Abbildung 3.45 [Farin 1996; Amidror 2002], da sich die Flächen durch die Determinanten mit $A_1 = \frac{1}{2}D_1$, $A_2 = \frac{1}{2}D_2$ und $A_3 = \frac{1}{2}D_3$ bestimmen lassen ($A = A_1 + A_2 + A_3$).

Die Interpolation erfolgt aus der Barizentrischen Koordinate $\overset{\circ}{\mathbf{x}}_i$ und den Tiefenwerten, welche in der Tiefenkarte in den Koordinaten $\mathbf{x}_{v+,i}$ gehalten sind, durch

$$\begin{aligned}Z_i &= [Z_{v+,1} \quad Z_{v+,2} \quad Z_{v+,3}] \\ &= \alpha_1 Z_{v+,1} + \alpha_2 Z_{v+,2} + \alpha_3 Z_{v+,3}.\end{aligned}\tag{3.90}$$

Die zu interpolierenden Pixel \mathbf{x}_i werden in der Tiefenkarte durch eine morphologische Closing-Operation (Kombination aus Dilatation \oplus und Erosion \ominus : $(\mathbf{M} \oplus \mathbf{S}) \ominus \mathbf{S}$) [Gonzalez u. Woods 2008] anhand einer Maske (Binärbild, \mathbf{M}), die die zur Verfügung stehenden Pixel \mathbf{x}_{v+} markiert, bestimmt. Das Strukturelement \mathbf{S} ist ein Quadrat mit der Kantenlänge b_i , welches die Grösse der Löcher zwischen den bereits vorhanden Pixel \mathbf{x}_{v+} bestimmt, die interpoliert werden. Aus der durch die Closing-Operation erzeugten Maske werden die zu interpolierenden Pixel \mathbf{x}_i extrahiert, von denen die Basisdaten \mathbf{x}_{v+} ausgeschlossen werden.

Bestimmung der Grauwerte im virtuellen Kamerabild

Zur Bestimmung der Grauwerte werden zunächst die 3D Daten \mathbf{X}_i aus den Pixeln \mathbf{x}_i der interpolierten Tiefenkarte mit der Gleichung C.6 aus dem Anhang C berechnet. Die Koordinate Z wird dabei nicht, wie in Gleichung C.6, durch die Disparität bestimmt, sondern ist der Wert Z_i aus der Tiefenkarte, welcher nach Gleichung 3.90 berechnet wurde. Die 3D Daten \mathbf{X}_i werden um die bereits vorhandenen Daten \mathbf{X}_{v+} zu \mathbf{X}_{i+} ergänzt. Die Koordinaten \mathbf{x}_{i+} , welche der Projektion von \mathbf{X}_{i+} auf die virtuelle Kamera entsprechen, ist die Pixelbasis \mathbf{x}_i kombiniert mit den Pixeln \mathbf{x}_{v+} , deren Koordinaten bereits bekannt sind.

Anschließend werden die Punkte \mathbf{X}_{i+} aus dem Kamerakoordinatensystem \mathcal{K}_V der virtuellen Kamera durch

$$\mathbf{X}_{i,d} = \mathbf{R}_V^T (\mathbf{X}_{i+} - \mathbf{t}_V) \quad (3.91)$$

in das System der Tiefenkamera (\mathcal{K}_1 in Abbildung 3.46) rücktransformiert. Die Punkte $\mathbf{X}_{i,d}$ werden dann mit Gleichung 3.8 aus Kapitel 3.3.2 in das jeweilige Grauwertbild projiziert und ergeben mit Berücksichtigung der Verzerrung nach Gleichung 3.13 aus Kapitel 3.3.2 die Koordinaten $\mathbf{x}_{i,k}$. Die Grauwerte $G_{i,k}$ an den jeweiligen Punkten $\mathbf{x}_{i,k}$ im Grauwertbild werden nun in das virtuelle Kamerabild an die Position des Pixels \mathbf{x}_{i+} übertragen.

Abbildung 3.46 stellt die Schritte zur Bestimmung des virtuellen Bilds vereinfacht in einem Diagramm dar. Deutlich erkennbar sind die zwei Kamerasysteme \mathcal{K}_1 (blau hinterlegt), welche der Kalibrierung einer der drei Kameras (Farbkamera der Kinect oder eine der zwei Industriekameras) mit der Tiefenkamera entspricht, und \mathcal{K}_V (magenta hinterlegt), welches das Kamerasystem der virtuellen Kamera darstellt.

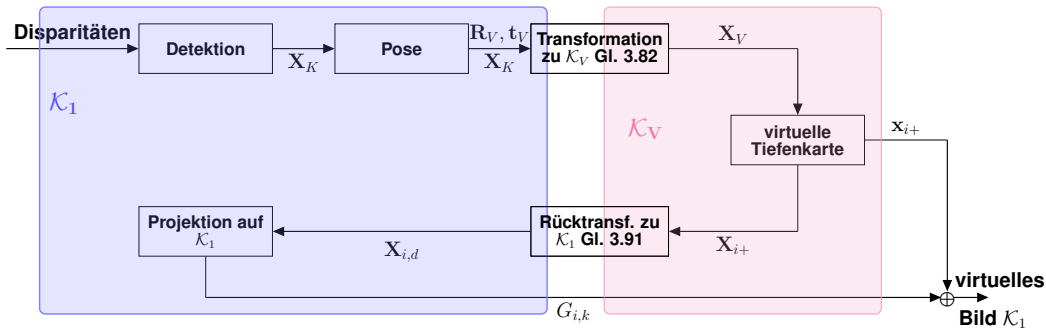


Abbildung 3.46.: Diagramm zur Bestimmung des virtuellen Bilds

Die *Detektion* (Kapitel 3.4) und die Posenbestimmung (Kapitel 3.5.1 und 3.5.2) werden im Kamerasystem der Tiefenkamera \mathcal{K}_1 ausgeführt. Anschließend werden die detektierten Daten \mathbf{X}_K mit der Gleichung 3.82 in das virtuelle Kamerasystem \mathcal{K}_V transformiert \mathbf{X}_V , um dort die virtuelle Tiefenkarte zu erzeugen und zu inter-

3. Material und Methoden

polieren (Unterkapitel *Ausbessern der virtuellen Tiefenkarte*). Dabei werden die 3D Daten \mathbf{X}_{i+} der virtuellen Tiefenkarte (Vereinigung der interpolierten Daten mit \mathbf{X}_V) bestimmt und gleichzeitig die dazugehörigen Pixelkoordinaten \mathbf{x}_{i+} , welche den Pixelkoordinaten auf dem zu erzeugenden virtuellen Bild entsprechen, zurückgelegt. Die 3D Punkte \mathbf{X}_{i+} werden nun mit der Gleichung 3.91 zurück in das Koordinatensystem \mathcal{K}_1 transformiert $\mathbf{X}_{i,d}$, um die Werte auf das originale Grauwertbild der Kamera, welche in \mathcal{K}_1 mit der Tiefenkamera kalibriert wurde, zu projizieren. Der Schritte der Projektion auf das Grauwertbild der Kamera in \mathcal{K}_1 ist in Abbildung 3.46 zur besseren Übersicht vereinfacht dargestellt. Die beschriebene Projektion in der Abbildung 3.46 berücksichtigt das Grauwertbild (Eingang des Grauwertbildes nicht dargestellt) und die auf dieses projizierten Pixelkoordinaten $\mathbf{x}_{i,k}$, um die Grauwerte $G_{i,k}$ an den Positionen $\mathbf{x}_{i,k}$ zu extrahieren und weiterzugeben. Zum Generieren des virtuellen Bilds werden die Grauwerte $G_{i,k}$ an die zuvor zurückgelegten Pixelkoordinaten \mathbf{x}_{i+} im virtuellen Bild geschrieben. Das virtuelle Bild berücksichtigt die Kamerakoordinatensysteme \mathcal{K}_1 und \mathcal{K}_V und kombiniert die Informationen beider zu einem Bild, so dass es nicht \mathcal{K}_1 oder \mathcal{K}_V zugeordnet werden kann.

Abbildung 3.47 zeigt an einem Beispiel die Grauwertbilder der virtuellen Kameraansicht in Abbildung 3.47b ohne die Ausbesserung der Tiefenkarte (mit Lücken in schwarz) und mit der interpolierten Tiefenkarte in Abbildung 3.47c. Zudem sind die jeweiligen Tiefenkarten in Abbildung 3.47e bzw. Abbildung 3.47f farbkodiert dargestellt. Die ursprüngliche Tiefeninformation ist in Abbildung 3.47d und das Grauwertbild der Farbkamera in Abbildung 3.47a gezeigt.

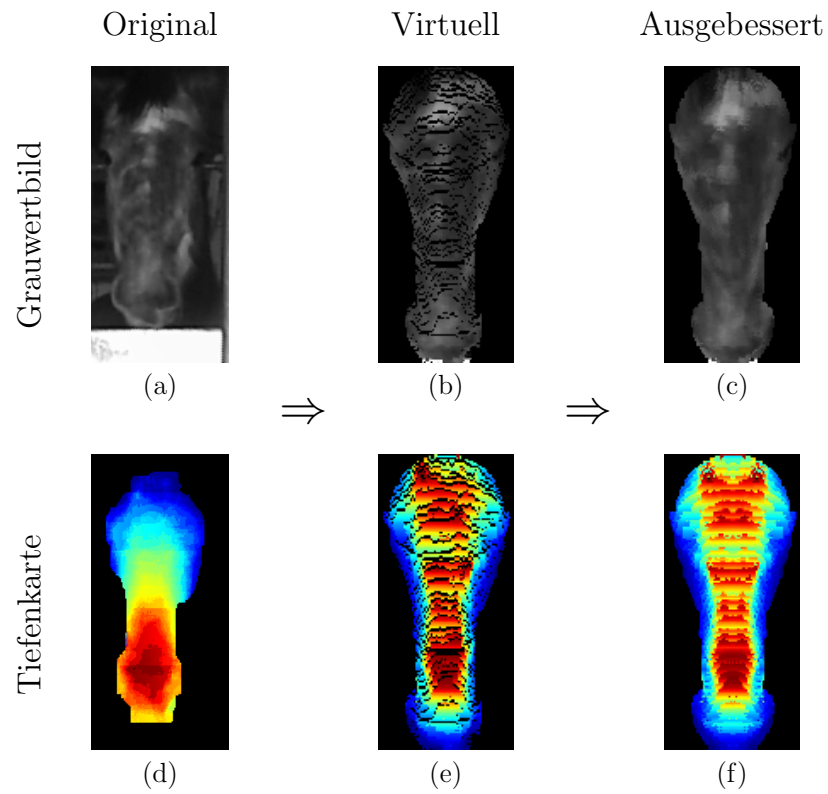


Abbildung 3.47.: Virtuelles Kamerabild mit Tiefenkarten [a+d) Original
b+e) ohne Korrektur c+f) mit Korrektur]

Transformation zur Bestimmung weiterer Kamerabilder

Die verwendete *Kamerakalibrierung* nach [Herrera u. a. 2011] ist in Kapitel 3.3.2 beschrieben und wurde einzeln für jede Kamera (Farbkamera der Kinect, sowie beide Industriekameras) mit der Tiefenkamera der Kinect durchgeführt (bezeichnet mit \mathcal{K}_i). Dadurch, dass die Kameras jeweils einzeln mit der Tiefenkamera kalibriert wurden, kann es zu unterschiedlichen intrinsischen Parametern β, γ bzw. \mathbf{K} kommen. Die intrinsischen Parameter β, γ werden zur Umrechnung der Disparitätenwerte d in die Tiefeninformation Z in Gleichung 3.17 (Kapitel 3.3.2) verwendet.

Um die entsprechenden virtuellen Kameraansichten zu bestimmen, lassen sich die *Detektion* aus Kapitel 3.4 und die Posenbestimmung aus Kapitel 3.5.1 bzw. 3.5.2 einzeln auf die jeweiligen Bilddaten anwenden. Alternativ ist die Transformation der virtuellen 3D Punkte $\mathbf{X}_{i,d}$ (\mathbf{X}_{i+} transformiert nach \mathcal{K}_1) von einer Kamerakalibrierung \mathcal{K}_1 in die jeweilige Kalibrierung der zweiten Kamera \mathcal{K}_2 möglich. Dies macht die wiederholte *Detektion* und Posenbestimmung überflüssig. Zudem kann die bereits aufgestellte Projektionsmatrix der virtuellen Kamera weiterhin benutzt werden, da die Projektion der Punkte \mathbf{X}_{i+} auf die Koordinaten \mathbf{x}_{i+} unberührt bleibt. Lediglich die Bestimmung der Grauwerte verändert sich zur Vorgehensweise des vorherigen Unterkapitels (siehe Abbildung 3.46). Die Punkte $\mathbf{X}_{i,d}$ werden nach der Transformation von \mathcal{K}_V nach \mathcal{K}_1 (Gleichung 3.91), bevor sie in das Kamerabild der zweiten Kamera projiziert werden, in die Kamerakalibrierung der zweiten Kamera \mathcal{K}_2 transformiert. Die Erweiterung des Vorgehens ist in Abbildung 3.48 dargestellt. Neben der Kamerakalibrierung \mathcal{K}_1 (blau) und \mathcal{K}_V (rot), welche auch in Abbildung 3.46 zu finden sind, ist die Kamerakalibrierung der zweiten Kamera \mathcal{K}_2 in Grün hinterlegt.

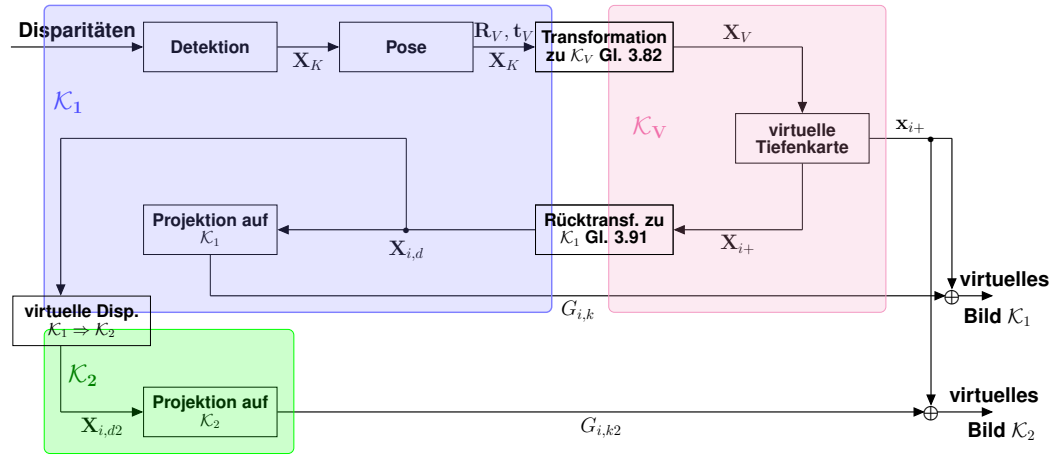


Abbildung 3.48.: Diagramm zur Bestimmung mehrerer virtuellen Bilder

Gemeinsamer Nenner der Kamerakalibrierungen \mathcal{K}_1 und \mathcal{K}_2 sind die Disparitätenwerte selbst, weshalb von den Punkten $\mathbf{X}_{i,d}$ nach der Kamerakalibrierung \mathcal{K}_1 zunächst virtuelle Disparitätenwerte erzeugt werden. Dazu werden die Punkte $\mathbf{X}_{i,d} = [X_{i,d} \ Y_{i,d} \ Z_{i,d}]^T$ nach Gleichung 3.5 (Kapitel 3.3.2) auf die Kameraebene

der Tiefenkamera projiziert

$$\begin{aligned} u_{d,v} &= \alpha_{x,1} \frac{X_{i,d}}{Z_{i,d}} + x_{0,1} \\ v_{d,v} &= \alpha_{y,1} \frac{Y_{i,d}}{Z_{i,d}} + y_{0,1} \end{aligned} \quad (3.92)$$

und die virtuellen Disparitätenwerte $d_{d,v}$ durch die Umkehr der Gleichung 3.17 aus Kapitel 3.3.2 nach

$$d_{d,v} = \frac{1}{Z_{i,d}\beta_1} + \gamma_1 \quad (3.93)$$

bestimmt. Die Weltkoordinaten der Kamerakalibrierung entsprechen den Kamerakoordinaten der Tiefenkamera, entsprechend werden keine extrinsischen Parameter zur Transformation benötigt.

Die virtuellen Disparitäten $d_{d,v}$ werden anschließend nach der Gleichung C.6 zur umgekehrten Projektion aus Anhang C direkt in die Kamerakalibrierung \mathcal{K}_2 transformiert. Daher ergeben sich die transformierten Koordinaten $\mathbf{X}_{i,d2} = [X_{i,d2} \ Y_{i,d2} \ Z_{i,d2}]^T$ zu

$$\begin{aligned} Z_{i,d2} &= \frac{1}{\beta_2(d_{d,v} - \gamma_2)} , \\ X_{i,d2} &= \frac{u_{d,v} - x_{0,2}}{\alpha_{x,2}} Z_{i,d2} \quad \text{und} \\ Y_{i,d2} &= \frac{v_{d,v} - y_{0,2}}{\alpha_{y,2}} Z_{i,d2} . \end{aligned} \quad (3.94)$$

Die Gleichungen 3.92 und 3.94 lassen sich zu der Transformation mit

$$\begin{aligned} Z_{i,d2} &= \frac{\beta_1}{\beta_2} \frac{1}{\frac{1}{Z_{i,d}} + \beta_1(\gamma_1 - \gamma_2)} , \\ X_{i,d2} &= \frac{\alpha_{x,1} \frac{X_{i,d}}{Z_{i,d}} + x_{0,1} - x_{0,2}}{\alpha_{x,2}} Z_{i,d2} \quad \text{und} \\ Y_{i,d2} &= \frac{\alpha_{y,1} \frac{Y_{i,d}}{Z_{i,d}} + y_{0,1} - y_{0,2}}{\alpha_{y,2}} Z_{i,d2} \end{aligned} \quad (3.95)$$

zusammenfassen. Die auf die Transformation der Punkte $\mathbf{X}_{i,d}$ zu $\mathbf{X}_{i,d2}$ folgende Projektion auf die Kamera der Kamerakalibrierung \mathcal{K}_2 erfolgt wie im vorherigen Unterkapitel *Bestimmung der Grauwerte im virtuellen Kamerabild* beschrieben und ergibt entsprechend das virtuelle Kamerabild der zweiten Kamera (\mathcal{K}_2).

3.6. Identifikation der einzelnen Individuen bei Pferden

Jafri u. Arabnia [2009] stellt eine Übersicht der unterschiedlichen Identifikationsverfahren zur Gesichtserkennung beim Menschen zusammen. Unter den Verfahren sind die ganzheitlichen (globalen) Verfahren, wie die Methode der Eigengesichter (PCA - *principal component analysis* - Hauptkomponentenanalyse) [Turk u. Pentland 1991b, a], der Fisherfaces (LDA - Lineare Diskriminanzanalyse) [Fisher 1936] und die Unabhängigkeitsanalyse (ICA - *independent component analysis*) [Comon 1994]. Neben den globalen Verfahren, werden in Jafri u. Arabnia [2009] auch die lokalen Verfahren erwähnt. In dieser Arbeit wird die Methode der Eigengesichter aus Kapitel 3.6.2 eingesetzt, welches ein globales Verfahren ist. In Kapitel 5.5 finden sich weitere Erläuterungen und die Diskussion zur Wahl eines globalen Identifikationsverfahrens und spezieller zur gewählten Methode PCA, deren Ergebnisse im Vergleich zu den Fisherfaces häufig schlechter ausfallen.

3.6.1. Vorverarbeitung durch Normalisierung der Grauwerte

Bereits im Kapitel 3.2 wurde beim Aufbau von *Versuchsstall* / *Messanordnung* darauf geachtet, eine möglichst kontrollierte Beleuchtungssituation innerhalb des Aufnahmebereichs mit geringem Einfall von (unkontrollierbarem) Aussenlicht zu schaffen. Obwohl eine homogene Ausleuchtung der Objekte in idealisierten Posen durch eine großflächig diffuse, künstliche Beleuchtung erzeugt werden kann, verändert sich der Lichteinfall und damit die Grauwerte, die auf dem Objekt in unterschiedlicher Entfernung erfasst werden.

Die in den realen Bilddaten entstehenden Abweichungen der Grauwerte unterschiedlicher Frames des selben Objekts werden durch die Normalisierung der Grauwerte in den Wertebereichen angepasst. Im Idealfall werden durch die Normalisierung identische Grauwerte auf unterschiedlichen Bildern des selben Objekts erzeugt, so dass die Abweichungen minimal ausfallen.

Die Grauwerte werden zur Normalisierung durch eine Histogrammäqualisation [Gonzalez u. Woods 2008, S. 144] angepasst, so dass die Grauwerte G_i des Originalbilds auf den maximalen Wertebereich $[0 \dots G_{max}]$ verteilt werden. Die Grauwerte G_i werden durch eine Transferfunktion $\mathcal{T}(G_i)$ auf die normalisierten Grauwerte $G_{N,i} = \mathcal{T}(G_i)$ abgebildet. Ziel der Histogrammäqualisation ist durch die Transformation ein normalisiertes Bild mit gleichverteiltem Histogramm zu erzeugen. Zunächst wird dazu eine Wahrscheinlichkeitsdichtefunktion $p(G_i)$ (PDF - *probability density function*) aufgestellt, welche für den diskreten Fall aus dem Histogramm erzeugt wird

$$p(G_i) = \frac{N_{G_i}}{N} . \quad (3.96)$$

N_{G_i} gibt dabei die Anzahl des Grauwertes G_i aus dem Histogramm des Originalbilds an und N die Gesamtzahl der Grauwerte im Bild. Dadurch gibt $p(G_i)$ die Häufigkeit ($p(G_i) \in [0 \dots 1]$) des Grauwertes G_i im Bild an. Aus der Wahrscheinlichkeitsdichte-

3.6. Identifikation der einzelnen Individuen bei Pferden

funktion $p(G_i)$ wird eine Summenverteilungsfunktion (CDF - *cumulative distribution function*)

$$c(G_i) = \sum_{i=0}^{G_i} p(G_i) \quad (3.97)$$

erstellt. Die Transferfunktion $\mathcal{T}(G_i)$ wird aus der Summenverteilung $c(G_i)$ und dem maximalen Grauwert G_{max} durch

$$\mathcal{T}(G_i) = G_{max} c(G_i) \quad (3.98)$$

aufgestellt. Die Kombination der Gleichungen 3.96, 3.97 und 3.98 ergibt

$$\mathcal{T}(G_i) = G_{max} \sum_{i=0}^{G_i} \frac{N_{G_i}}{N} = \frac{G_{max}}{N} \sum_{i=0}^{G_i} N_{G_i} . \quad (3.99)$$

Die Transferfunktion $\mathcal{T}(G_i)$ streckt solche Grauwertbereiche im Histogramm, die eine hohe Auftrittswahrscheinlichkeit ($p(G_i) > \frac{1}{G_{max}}$) im Bild aufweisen, und staucht die Bereiche mit einer geringen Häufigkeit ($p(G_i) < \frac{1}{G_{max}}$). Die Transferfunktion $\mathcal{T}(G_i)$ der Grauwerte zur Histogrammäqualisation ist monoton steigend und erhöht den Kontrast im transformierten Bild. Ein entsprechendes Vorgehen zur Normalisierung vor der Identifikation ist, neben weiteren Verfahren, in den Arbeiten von Delac u. a. [2005] und Ramírez-Gutiérrez u. a. [2010] beschrieben.

Krutsch u. Tenorio [2011] stellt eine leicht veränderte Transferfunktion auf

$$\mathcal{T}(G_i) = \text{round}\left(\frac{G_{max}}{N - N_0} \left(\sum_{i=0}^{G_i} N_{G_i} - N_0\right)\right) = \text{round}\left(\frac{G_{max}}{N - N_0} \sum_{i=1}^{G_i} N_{G_i}\right) , \quad (3.100)$$

in welcher der minimale Wert der CDF aus Gleichung 3.97 unberücksichtigt bleibt. Das in dieser Arbeit verwendete Verfahren zur Normalisierung der Grauwerte entspricht, bis auf die Rundung, der Gleichung 3.100, welche durch eine Abrundung ersetzt wird:

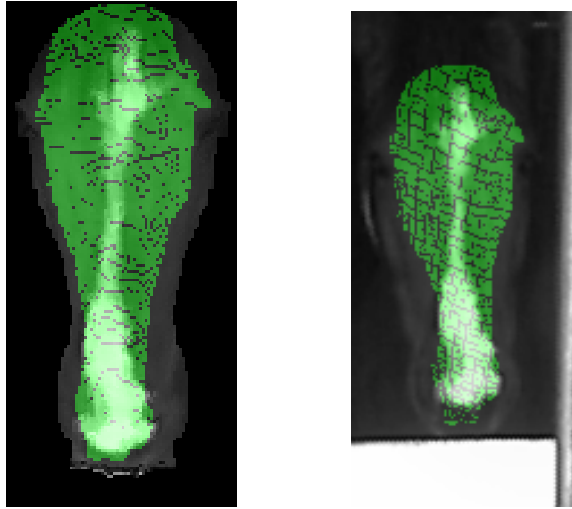
$$\mathcal{T}(G_i) = \left\lfloor \frac{G_{max}}{N - N_0} \sum_{i=1}^{G_i} N_{G_i} \right\rfloor . \quad (3.101)$$

Die Transferfunktion aus Gleichung 3.101, welche zur Normalisierung der Grauwerte vor der Identifikation angewendet wird, ist nicht aus den kompletten Grauwerten des virtuellen Kamerabilds aus Kapitel 3.5.3 bestimmt. Das Berücksichtigen aller Werte im virtuellen Bild hätte zur Folge, dass die Werte im Randbereich der Pferdeköpfe und die interpolierten Grauwerte Einfluss auf die Transformation haben, welches verhindert werden sollte.

Das virtuelle Kamerabild wurde in Kapitel 3.5.2 aus den ursprünglichen Punkten \mathbf{X}_K des Pferdekopfes berechnet, die aus den Disparitätenwerten innerhalb der dilatierten Maske $\mathbf{M} \oplus \mathbf{S}$ bestimmt wurden. Zum Erstellen der Transferfunktion werden

3. Material und Methoden

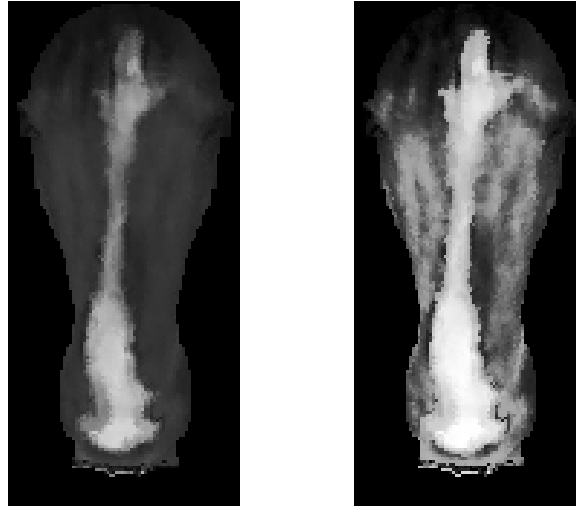
nur die Grauwerte berücksichtigt, welche den aus der Detektion (siehe Kapitel 3.4) stammenden Bereichen (Maske \mathbf{M} , nicht der dilatierten Maske $\mathbf{M} \oplus \mathbf{S}$ aus Kapitel 3.5.2) auf dem virtuellen Kamerabild entsprechen. Der entsprechende Bereich ist in Abbildung 3.34a aus Kapitel 3.5.2 auf dem Grauwertbild der Farbkamera in Weiß dargestellt. Abbildung 3.49 zeigt den entsprechenden Bereich auf dem virtuellen Kamerabild (3.49a) und auf dem Grauwertbild der Farbkamera (3.49b) in Grün. Die Lücken des grünen Bereichs in Abbildung 3.49b entstehen durch die Projektion auf das virtuelle Kamerabild, auf dem die Auswahl stattfindet.



(a) Virtuelles Kamerabild (b) Bild der Farbkamera

Abbildung 3.49.: Auswahl zur Bestimmung der Transferfunktion $\mathcal{T}(G_i)$

Die Transferfunktion wird anschließend zur Normalisierung auf das komplette Grauwertbild der virtuellen Kamera angewendet, welches durch den Einsatz einer Umsetzungstabelle geschieht (LUT - *lookup-table*). Abbildung 3.50 zeigt auf der linken Seite (Abb. 3.50a) das originale Bild der virtuellen Kamera und auf der rechten Seite (Abb. 3.50b) das durch die Histogrammäqualisation angepasste Grauwertbild.



(a) ohne Normalisierung (b) mit Normalisierung

Abbildung 3.50.: Virtuelles Kamerabild mit (b) und ohne (a) Normalisierung der Grauwerte

3.6.2. Eigengesichter auf 2D Daten

Gesichter lassen sich durch eine lineare Kombination von Eigengesichter (*eigenfaces*) darstellen [Sirovich u. Kirby 1987; Kirby u. Sirovich 1990]. Die in diesem Kapitel beschriebene Methode von Turk u. Pentland [1991b] verwendet die Eigengesichter zur Identifikation von Gesichtern.

Die Eigengesichter sind ein Satz von Merkmalsvektoren, welche die Varianz zwischen den Bildern der Gesichter in einem Lerndatensatz beschreiben. Die Eigengesichter werden durch die Hauptkomponentenanalyse (PCA - *principle component analysis*) aus mehreren Gesichtsbildern bestimmt. Im Mathematischen entsprechen die Eigengesichter den Eigenvektoren der Kovarianzmatrix (\mathbf{C}) eines Satzes von Lernbildern. Die Bilddaten werden in einem Vektor $\mathbf{\Gamma}$ gehalten, welcher alle Pixel im Bild $N = N_{\text{Zeilen}} \times N_{\text{Spalten}}$ in nur einer Spalte speichert. Über den Lerndatensatz mit M Bildvektoren $\mathbf{\Gamma}_1 \cdots \mathbf{\Gamma}_M$ wird der durchschnittliche Vektor $\mathbf{\Psi} = \frac{1}{M} \sum_{n=1}^M \mathbf{\Gamma}_n$ bestimmt, um mittelwertfreie Bildvektoren zu erhalten $\mathbf{\Phi}_i = \mathbf{\Gamma}_i - \mathbf{\Psi}$. Aus den Vektoren $\mathbf{\Phi}$ wird eine Matrix $\mathbf{A} = [\mathbf{\Phi}_1, \cdots \mathbf{\Phi}_M]$ erstellt, um die Kovarianzmatrix

$$\mathbf{C} = \frac{1}{M} \sum_{n=1}^M \mathbf{\Phi}_n \mathbf{\Phi}_n^T = \mathbf{A} \mathbf{A}^T \quad (3.102)$$

zu bilden. Der Vektor \mathbf{u}_k sei ein Eigenvektor und der Wert λ_k der entsprechende Eigenwert der Kovarianzmatrix \mathbf{C} ($\mathbf{C} \mathbf{u}_k = \lambda_k \mathbf{u}_k$). Die Eigenvektoren \mathbf{u}_k (Eigengesichter) spannen dabei einen Raum auf, welcher als Gesichtsraum bezeichnet wird.

3. Material und Methoden

In diesem Raum lässt sich jedes Gesicht als ein Punkt $\mathbf{\Omega} = [\omega_1, \dots, \omega_M]^T$ beschreiben.

Da die Anzahl der Bilder im Lerndatensatz M häufig deutlich kleiner ist als die Anzahl der Pixel in den Bildern N , sind nur $M - 1$ Eigenwerte λ_k ungleich Null. Zur Berechnung der Eigenvektoren \mathbf{u}_k kann daher die Matrix $\mathbf{L} = \mathbf{A}^T \mathbf{A}$ genutzt werden. Die ersten M Eigenwerte (alle $\lambda_k = 0$ von $k > M$) von \mathbf{L} und \mathbf{C} sind identisch, die Eigenvektoren \mathbf{v}_k von \mathbf{L} lassen sich durch

$$\mathbf{u}_k = \mathbf{A} \mathbf{v}_k \quad (3.103)$$

in die Eigenvektoren \mathbf{u}_k von \mathbf{C} transformieren (siehe Turk u. Pentland [1991b], ähnlich der Singulärwertzerlegung in [Bronštejn u. a. 2012, S. 328, Gl. 4.218 a/b]). Die Eigenwerte λ_k erlauben es die Eigenvektoren \mathbf{u}_k nach ihrem Einfluss in den Bildern des Lerndatensatzes zu sortieren. Mit dem vollen Satz der M Eigenvektoren \mathbf{u}_k (Eigengesichter) lassen sich die Bilder des Lerndatensatzes komplett rekonstruieren. Zur Identifikation reicht jedoch eine kleinere Menge an Eigengesichtern $M' < M$ aus, welche den M' größten Eigenwerten entsprechen. Der Gesichtsraum, welcher durch die M' Eigenvektoren aufgespannt wird, ist entsprechend kleiner.

Ein Gesicht $\mathbf{\Gamma}$ wird in den verkleinerten Gesichtsraum mit der Dimension M' durch

$$\omega_k = \mathbf{u}_k^T (\mathbf{\Gamma} - \mathbf{\Psi}) \quad (3.104)$$

projiziert und in diesem durch den Punkt $\mathbf{\Omega}' = [\omega'_1, \dots, \omega'_{M'}]^T$ repräsentiert.

Um Gesichter zu identifizieren werden zuvor die einzelnen Bilder des Lerndatensatzes entsprechend der Gleichung 3.104 in den reduzierten Gesichtsraum projiziert und als Punkte $\hat{\mathbf{\Omega}}'_i = [\hat{\omega}'_{i,1} \dots \hat{\omega}'_{i,M'}]^T$ mit $i \in [1 \dots M]$ gespeichert. Den einzelnen Punkten $\hat{\mathbf{\Omega}}'_i$ kann durch ein Label L_i der Lerndatensätze ein Individuum zugeordnet werden. Anschließend wird das zu überprüfende Bild, welches auf den Punkt $\mathbf{\Omega}'$ projiziert wurde, mit den Punkten $\hat{\mathbf{\Omega}}'_i$ des Lerndatensatzes verglichen. Zum Vergleich wird die Euklidische Distanz $d^2(\mathbf{\Omega}', \hat{\mathbf{\Omega}}'_i)$, welche auf der Euklidischen Norm [Bronštejn u. a. 2012, S. 283] (L_2 -Norm) basiert, nach

$$d^2(\mathbf{\Omega}', \hat{\mathbf{\Omega}}'_i) = \sqrt{\sum_{c=1}^{M'} (\omega'_c - \hat{\omega}'_{i,c})^2}. \quad (3.105)$$

berechnet. Zur Identifikation wird die minimale Distanz d_{min}^2 aus allen Distanzen $d^2(\mathbf{\Omega}', \hat{\mathbf{\Omega}}'_i)$ zu den Lerndaten $\hat{\mathbf{\Omega}}'_i$ bestimmt. Ist d_{min}^2 kleiner als ein zuvor festgelegter Schwellwert Θ_d , dann wird das zu prüfende Gesicht dem Lerndatensatz i (mit dem Punkt $\hat{\mathbf{\Omega}}'_i$ im Gesichtsraum) zugeordnet für den die Distanz $d^2(\mathbf{\Omega}', \hat{\mathbf{\Omega}}'_i)$ minimal war. Über das entsprechende Label L_i des Lerndatensatzes i wird das entsprechende Individuum bestimmt.

Ist die minimale Distanz $d_{min}^2 \geq \Theta_d$ dem Schwellwert, so gilt das Gesicht als unerkannt.

Die Rückprojektion des Punktes $\mathbf{\Omega}' = [\omega'_1, \dots, \omega'_{M'}]^T$

$$\mathbf{\Gamma}' = [\mathbf{u}_1 \cdots \mathbf{u}_{M'}] \mathbf{\Omega}' + \mathbf{\Psi} = \sum_{c=1}^{M'} \omega'_c \mathbf{u}_c + \mathbf{\Psi} \quad (3.106)$$

ergibt eine Rekonstruktion $\mathbf{\Gamma}'$ des originalen Gesichts $\mathbf{\Gamma}$ mit den Daten des reduzierten Gesichtsraumes.

Die Rekonstruktion $\mathbf{\Gamma}'$ wird durch Turk u. Pentland [1991b] dazu genutzt ein Bild eines Gesichtes von sonstigen Bildern zu unterscheiden. In dieser Arbeit wird die Rekonstruktion jedoch nur zur Veranschaulichung verwendet, da Kapitel 3.4.4 bereits die Detektion der Pferdeköpfe behandelt. Die Abbildung 3.51 zeigt die Rekonstruktion $\mathbf{\Gamma}'_1, \mathbf{\Gamma}'_2$ und $\mathbf{\Gamma}'_3$ (3.51 f-h) der drei originalen Bilder $\mathbf{\Gamma}_1, \mathbf{\Gamma}_2$ und $\mathbf{\Gamma}_3$ (3.51 a-c) mit nur einem Eigenface \mathbf{u}_1 (3.51 e).

Links neben dem Eigenface (3.51 e) ist der Vektor $\mathbf{\Psi}$ (3.51 e) gezeigt, dieser stellt das mittlere Bild der drei Eingangsbilder ($\mathbf{\Psi} = \frac{\mathbf{\Gamma}_1 + \mathbf{\Gamma}_2 + \mathbf{\Gamma}_3}{3}$) dar. Mit nur einem Eigenface lassen sich die Bilddaten $\mathbf{\Gamma}_1, \mathbf{\Gamma}_2, \mathbf{\Gamma}_3$ nicht zufriedenstellend rekonstruieren, welches vorrangig in der Rekonstruktion $\mathbf{\Gamma}'_3$ (3.51h) erkennbar ist. Die Rekonstruktion $\mathbf{\Gamma}'_3$ (3.51h) ähnelt eher dem Eingangsbild $\mathbf{\Gamma}_2$ (3.51b) als dem Bild $\mathbf{\Gamma}_3$ (3.51c).

Abbildung 3.52 zeigt die Rekonstruktion $\mathbf{\Gamma}'_1, \mathbf{\Gamma}'_2$ und $\mathbf{\Gamma}'_3$ (3.52 g-i) der drei originalen Bilder $\mathbf{\Gamma}_1, \mathbf{\Gamma}_2$ und $\mathbf{\Gamma}_3$ (3.52 a-c) mit zwei Eigengesichtern \mathbf{u}_1 (3.52 e) und \mathbf{u}_2 (3.52 f).

Das mittlere Bild der drei Eingangsbilder $\mathbf{\Psi}$ in Abbildung 3.52d entspricht dem aus Abbildung 3.51d. Zur Rekonstruktion der Bilder $\mathbf{\Gamma}_1, \mathbf{\Gamma}_2, \mathbf{\Gamma}_3$ wurde in Abbildung 3.52 lediglich ein weiteres Eigenface \mathbf{u}_2 verwendet, wodurch die Rekonstruktion $\mathbf{\Gamma}'_3$ Abbildung 3.52i dem Original $\mathbf{\Gamma}_3$ Abbildung 3.52c deutlich ähnlicher ist.

3. Material und Methoden

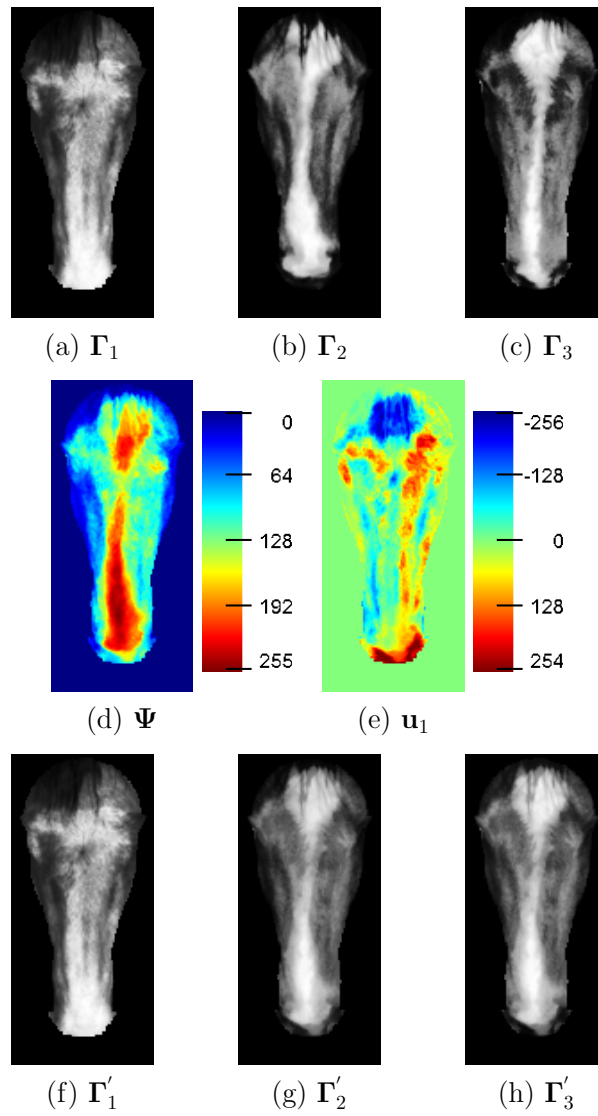


Abbildung 3.51.: Rekonstruktion mit einem Eigenface

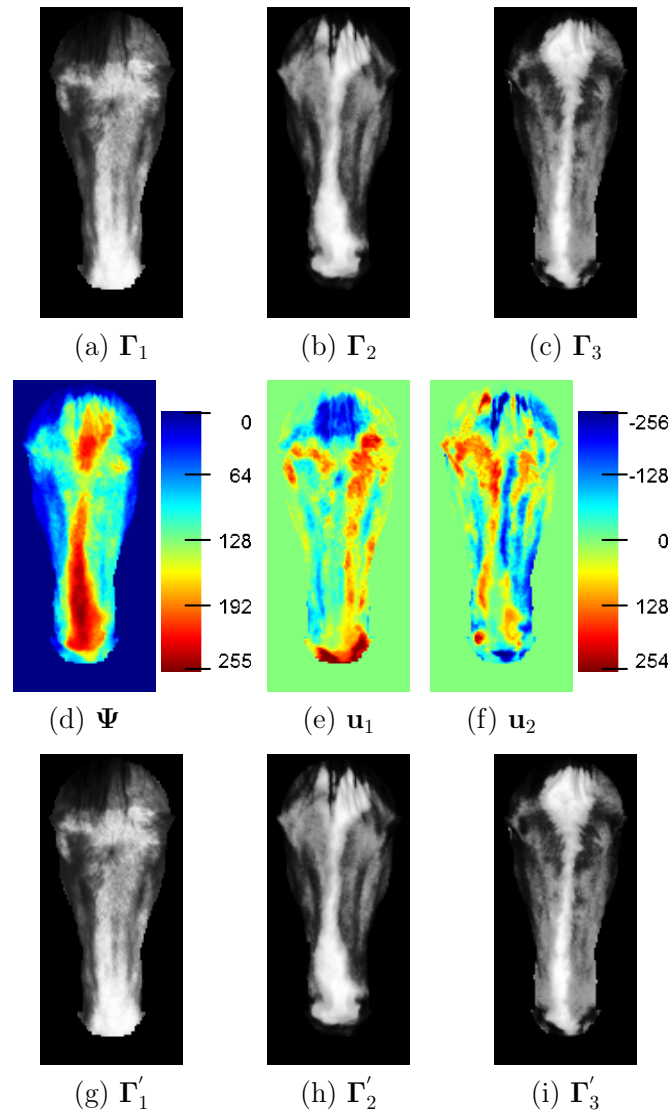


Abbildung 3.52.: Rekonstruktion mit zwei Eigengesichtern

3.7. Versuchsdurchführungen

3.7.1. Genauigkeitsmessung der Tiefendaten

Zur Untersuchung der Messgenauigkeit der Kinect, die zur Aufzeichnung der Disparitätenbilder eingesetzt wird, wurde ein möglichst planares Messobjekt aus unterschiedlichen Entfernungen aufgenommen. Aus den aufgezeichneten Disparitätenbildern lässt sich mit Hilfe einer zuvor durchgeführten Kalibrierung der Kamera (siehe Kapitel 3.3.2) ein 3D-Tiefenwert für jeden gültigen Pixel bestimmen. Im Unterkapitel *Tiefenmessung 1* werden die entsprechend gemessenen Tiefenwerte untersucht. Im Unterkapitel *Tiefenmessung 2* wird die Untersuchung der Messgenauigkeit an einem Kalibrierobjekt beschrieben.

Hardware Aufbau

In Abbildung 3.53 ist der Messaufbau skizziert dargestellt. Die im Messaufbau verwendete Verfahreinheit treibt dabei einen Schlitten, welcher auf einer Schiene läuft, durch einen Schrittmotor an. Der blaue Pfeil in Abbildung 3.53a/c deutet die Fahrtrichtung des Schlittens mit der Kamera auf der linearen Verfahreinheit an.

Der Schlitten wird durch einen Schrittmotor ILS1M853PB1A0 von Schneider Electric angetrieben. Dieser kann die Position des auf der Schiene verfahrenen Schlittens durch einen Inkrementalgeber grob bestimmen. Eine Position auf der Schiene ein zweites Mal „exakt“ anzufahren ist jedoch nur mit Hilfe des Inkrementalgebers durch dessen Schlupf nicht möglich. Um dies zu kompensieren wurde ein Magnetband auf der Schiene der Verfahreinheit montiert. Die aktuelle Position des Schlittens auf der Schiene kann so über einen Lesekopf am Schlitten ausgelesen werden. Die Position des Schlittens kann dadurch absolut mit einer Genauigkeit von $100\mu m$ bestimmt werden. Um die Positionswerte auch auf einem Rechner zur Verfügung zu haben, wurde die Magnetbandanzeige MA502 von SIKO verwendet.

Als Messobjekt wurde eine Stellwand verwendet, die auch auf Messen zum Einsatz kommt. Diese steht von alleine und bietet eine ausreichend große Fläche, die planar ist. Entsprechend Abbildung 3.53b hat die Fläche der Stellwand eine Höhe von $150,2\text{ cm}$ und eine Breite von $126,1\text{ cm}$. Wie in den Originalbildern des Aufbaus aus Abbildung 3.54b zu sehen, ist die Stellwand kurz oberhalb der Schiene der Verfahreinheit positioniert.

Die Kamera ist in der Versuchsanordnung auf dem Schlitten der linearen Verfahreinheit montiert. Dies hat gegenüber der umgekehrten Anordnung, den Vorteil, dass die Kamera eine deutlich geringere Masse und kleinere Abmessungen im Vergleich zum verwendeten Messobjekt hat. Hierdurch treten beim Verfahren weniger ungewollte Schwingungen und damit Messungenauigkeiten auf. Die Kamera wird ca. 40 cm über dem Schlitten durch Aluprofile gehalten. Dies ist in der Rückansicht Abbildung 3.54a und in der Seitenansicht Abbildung 3.54b des Aufbaus zu erkennen. Der Abstand zwischen Kamera und dem Schlitten wurde dabei gewählt, um auch bei größerer Entfernung keine Verdeckung der direkte Strecke zwischen Kamera und Messwand durch die Schiene der Verfahreinheit zu haben.

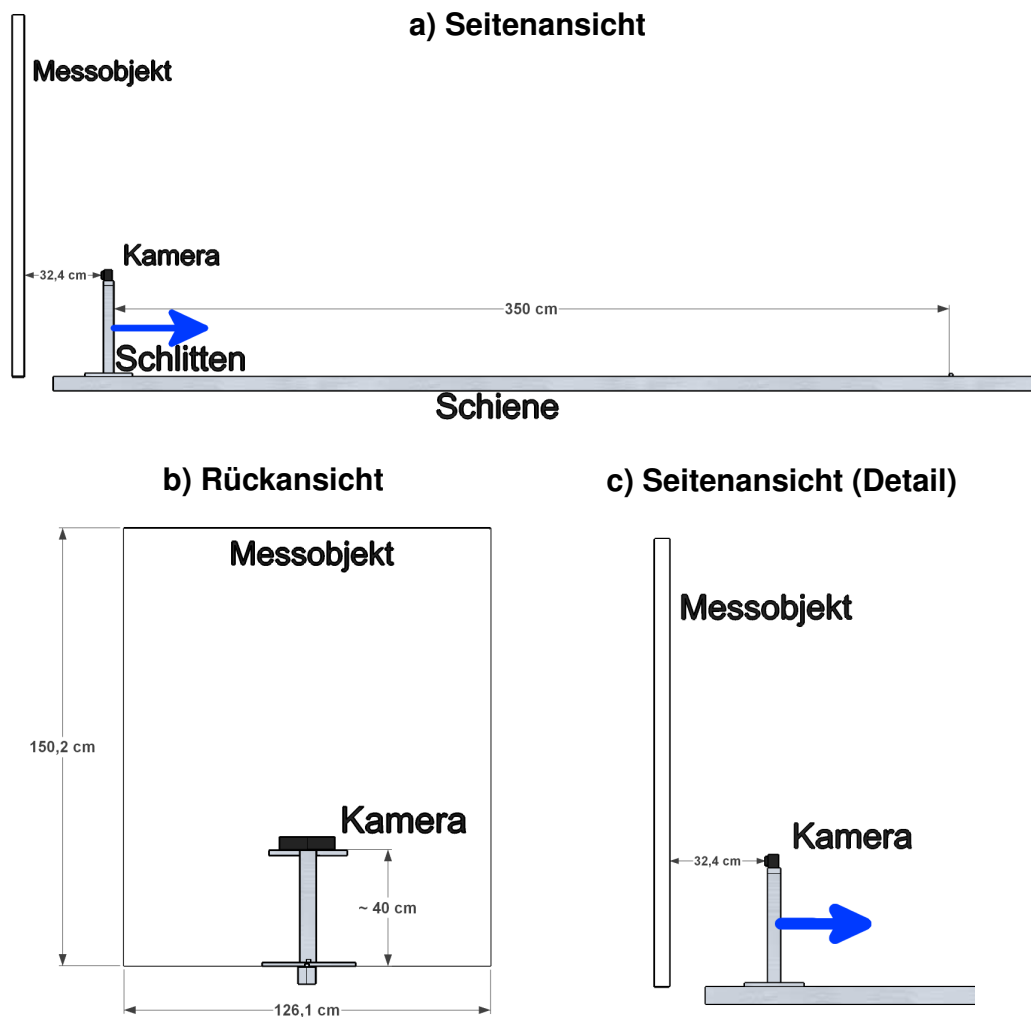


Abbildung 3.53.: Skizze der Messanordnung a) Seitenansicht, b) Rückansicht, c) Detailansicht

3. Material und Methoden

a) Rückansicht



b) Seitenansicht

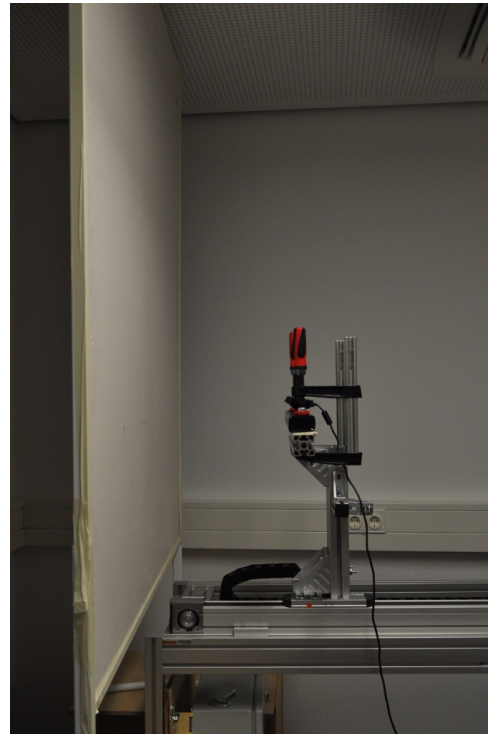


Abbildung 3.54.: Originalaufnahmen der Messanordnung a) Rückansicht, b) Seitenansicht

Tiefenmessung 1

Der Schlitten mit der Kamera wurde auf seine Nullposition auf der Schiene verfahren, dann wurde die Stellwand möglichst parallel zur Kameraebene aufgestellt. Dies entspricht lediglich einer groben Positionierung am Kameragehäuse, da die tatsächliche Lage des Sensorchips in der Kamera nicht berücksichtigt wurde. In der Nullposition des Schlittens wurde ein Abstand von $32,4\text{ cm}$ zwischen Messobjekt und Kameragehäuse gelassen. Ein direktes Positionieren der Kamera am Messobjekt ist aufgrund des Aufbaus der Verfahreinheit nur schwer möglich, zudem liefert die Kinect in einem Abstand unter 50 cm (Position 16 cm) keine verwertbaren Messergebnisse. In der Entfernung von $50 - 70\text{ cm}$ (Position $17 - 37\text{ cm}$) werden lediglich vereinzelt verwertbare Messergebnisse von der Kinect zurückgeliefert. Ab 70 cm (Position 38 cm) konnten alle Pixel auf dem zur Auswertung verwendeten Bildbereich verwendet werden.

Von der Nullposition (Position 0 cm) aus wurde der Schlitten auf einer Strecke von 350 cm in Schritten von 1 cm von dem Messobjekt weg verfahren (siehe Fahrtrichtung gekennzeichnet durch blauen Pfeil in Abbildung 3.53a/c). Die Software

der Verfahreinheit wurde dabei so eingestellt, dass gewartet wurde, bis die von der Magnetleiste abgelesenen Werte konstant blieben. In jedem Schritt wurden 200 Aufnahmen der Stellwand aufgenommen.

Zur Auswertung der Daten wurde eine Maske erstellt, welche einen Teil des Bildes abdeckt. Die Maske ist in Abbildung 3.55 gezeigt, sie ist in Grün auf dem Infrarot-Bild (aufgenommen an Position 350 cm mit veränderten Grauwerten, sowie mit dem Tiefenbild in Deckung gebracht) eingezeichnet, wobei das Messobjekt rot umrandet ist. Dieser Bereich des Bilds ist auf der gesamten Fahrt sichtbar, also auch aus der Entfernung an der Position 350 cm (Abstand zum Messobjekt: 382,4 cm). Zudem werden durch die Maske jeweils die gleiche Anzahl an Messwerten berücksichtigt. Die Maske wird entsprechend nicht dem Objekt auf der Kamera angepasst, welches aus größerer Entfernung kleiner erscheint.

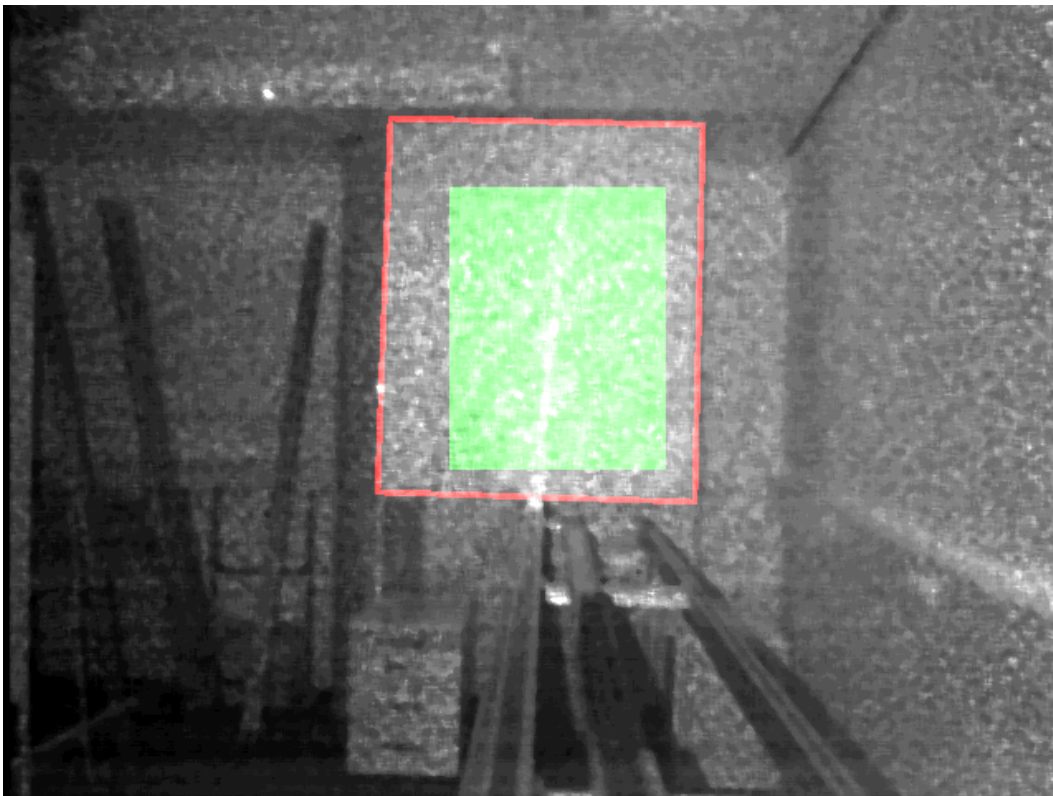


Abbildung 3.55.: Maske auf Infrarot-Bild an Position 350 cm (Messobjekt rot umrandet, Maske in grün)

Die Disparitätenwerte jeder Aufnahme, welche sich in der beschriebenen Maske befinden, wurden mit Hilfe der Kalibrierung aus Kapitel 3.3.2 und der im Anhang C beschriebenen Rekonstruktion in 3D-Punkte umgerechnet. Daraus ergibt sich eine

3. Material und Methoden

Punktwolke, welche sich, durch die Planarität des Messobjekts, auf einer Ebene befinden sollte. Daher wird für alle 200 Aufnahmen pro Position eine Ebene in die 3D-Punktdaten eingepasst. Hierdurch lässt sich eine abweichende Orientierung von Kamerasensor und Messobjekt kompensieren, welche physikalisch nur schwer auszurichten sind. Zuletzt wird die Abweichung jedes Punktes zur bestimmten Ebene berechnet, welche der kleinsten Distanz zwischen jedem Punkt und der bestimmten Ebene entspricht. Zur Auswertung werden sogenannte Boxplots eingesetzt, welche in Kapitel 3.8.1 beschrieben sind. Durch Sie werden die Fehler der 200 Aufnahmen in jeder Position repräsentiert. Die Ergebnisse der beschriebenen Aufnahmen sind im Kapitel 4.2 zu finden.

Tiefenmessung 2

Zur Aufnahme der Bilddaten wurde die Kamera entsprechend der Aufnahmen des vorherigen Unterkapitels *Tiefenmessung 1* auf der Lineareinheit verfahren. Mit dem Unterschied, dass eine Schrittweite von 20 cm, beginnend bei der Position 20 cm (Abstand zu Messobjekten $\sim 52,4$ cm) bis zur Position 360 cm, gewählt wurde. Zudem wurden pro Objekt und Position nur 5 Aufnahmen gemacht, wobei zu den Disparitätenbildern auch die RGB-Bilder (RGB-Bilder halten 3 Kanäle mit den Farben Rot, Grün und Blau) der Farbkamera mit aufgezeichnet wurden. Ein zusätzlich auf der Stellwand angebrachtes Kalibrierobjekt (Schachbrettmuster) ist nur auf den Farbbildern sichtbar. Auf den Disparitätenbildern ist das Schachbrett nicht erkennbar, da es plan auf die Stellwand geklebt ist. Insgesamt wurde das Messobjekt (Stellwand) in 5 unterschiedlichen Orientierungen zur Kamera aufgenommen. Hierzu wurde das Messobjekt aufgestellt, um dann die unterschiedlichen Positionen auf der Lineareinheit nacheinander anzufahren. Erst nachdem die Kamera, bedingt durch die erstellte Software, an ihre Ursprungsposition (Position 20 cm) gefahren ist, wurde das Messobjekt erneut umgestellt. Die unterschiedlichen Orientierungen der Stellwand zur Kamera sind im untersten Bereich der Abbildung 3.56 gezeigt und mit Objektnummern für jede Orientierungsänderung des Messobjekts zur Kamera beschriftet. Ein genauer Abstand von Messobjekt und Kamera wurde nicht ermittelt. Dieser ist, nicht nur durch die bereits erwähnte unbekannte Lage der Kamerasensoren in der Kamera, sondern auch durch die unterschiedlichen Ausrichtungen der Stellwand zur Kamera, nur schwer festzulegen. Die Abbildung 3.56 zeigt auf der linken Seite die unterschiedlichen Positionen von 40 cm bis 140 cm, welche für das dritte Objekt angefahren wurden. Der angegebene Abstand ist eine grobe Angabe eines Punktes auf dem Schachbrett, da beim Umstellen versucht wurde den Abstand von Kamera und Schachbrett in etwa konstant zu halten. Wie bereits in dem vorherigen Unterkapitel *Tiefenmessung 1* wurde die Position 20 cm, obwohl sie aufgenommen wurde, nicht weiter berücksichtigt, da der geringe Abstand keine verwertbaren Tiefeninformationen liefert.

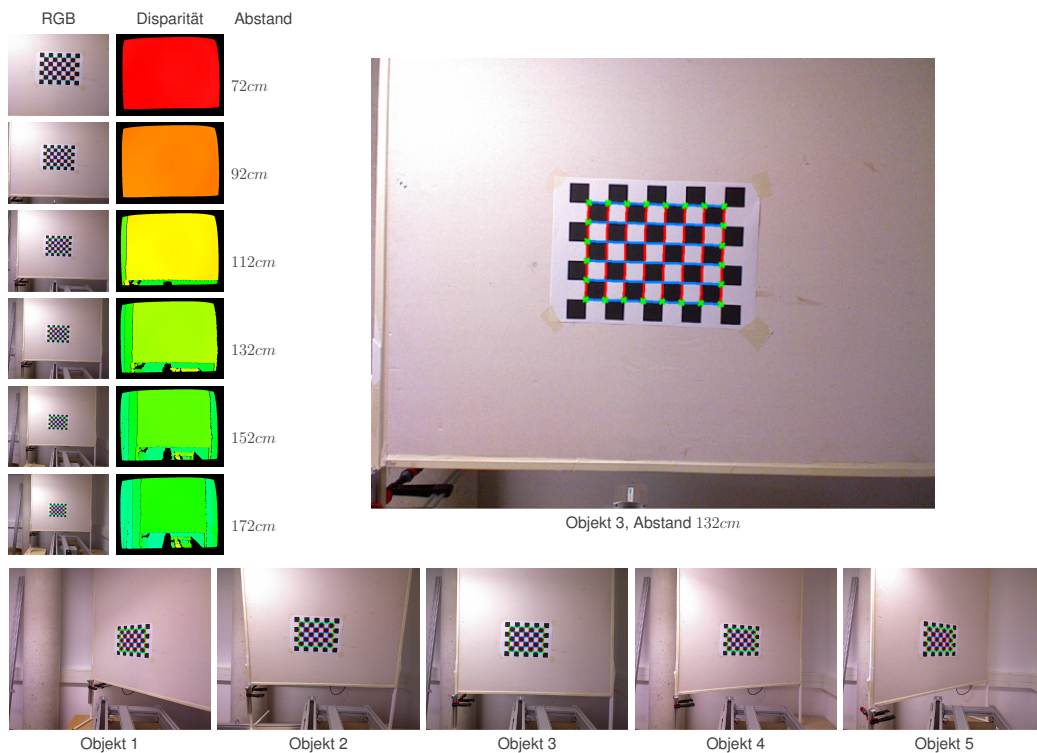


Abbildung 3.56.: Messablauf

Oben rechts in Abbildung 3.56 ist das Kalibrierobjekt (Schachbrett) auf Objekt 3 an Position 100 cm (entspricht Abstand $\sim 132\text{ cm}$) gezeigt, dies wurde durch die Farbkamera der Tiefenkamera aus Kapitel 3.3 aufgenommen. Bei einem vorher definierten Kalibrierobjekt sind die Entfernungen (blaue und rote Linien auf dem Schachbrett) zwischen den äußeren Punkten des Schachbretts bekannt. Die auf dem Kalibrierobjekt bekannten Entfernungen können mit den durch das Disparitätenbild gemessenen Distanzen verglichen werden. Jedoch ist, wie bereits erwähnt, das Kalibrierobjekt und auch dessen Muster, die einzelnen Felder des Schachbretts, nicht auf dem Disparitätenbild zu erkennen. Daher wird zunächst aus jedem Disparitätenwert, nach dem im Anhang C beschriebenen Verfahren, ein 3D-Punkt im Raum berechnet. Dieser wird anschließend entsprechend der Gleichung 3.8 aus Kapitel 3.3.2 auf die Kameraebene der Farbkamera projiziert, welches in Abbildung 3.57 als Transformation auf das RGB-Bild beschrieben ist. Da die Markerpunkte des Kalibrierobjekts auf dem Farbbild bestimmt werden können, ist eine Zuordnung der durch die Transformation auf jene Punkte abgebildeten 3D-Punkte möglich. Es werden entsprechend nur solche 3D-Punkte weiter behandelt, welche durch die Abbildung auf das Farbbild mit den extrahierten Punkten des Schachbretts (Kalibrierobjekt) übereinstimmen. Die zur Projektion und Rekonstruktion nötige *Kamerakalibrierung* wurde mit 24 zufällig gewählten Bildpaaren (Farbbild und Disparitätenbild) der Messreihe durchgeführt.

3. Material und Methoden

Zuletzt kann der Abstand dieser 3D-Punkte zueinander bestimmt werden, welche dann mit den Abmessungen des Kalibrierobjekts verglichen werden können. Hierzu wird die Distanz zwischen jeweils zwei der äußeren Markerpunkte (grüne Kreuze rechts in Abbildung 3.57) im \mathbb{R}^3 in horizontaler bzw. vertikaler Richtung berechnet. Die Distanz l_m des ersten Punktes $\mathbf{X}_1 = [X_1 \ Y_1 \ Z_1]^T$ zum zweiten Punkt $\mathbf{X}_2 = [X_2 \ Y_2 \ Z_2]^T$ wird entsprechend durch die Euklidische Norm [Bronštejn u. a. 2012, S. 283] (L_2 -Norm) aus der Differenz $\mathbf{X}_1 - \mathbf{X}_2$ der beiden Punkte berechnet:

$$l_m = \|\mathbf{X}_1 - \mathbf{X}_2\| = \sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2 + (Z_1 - Z_2)^2} \quad (3.107)$$

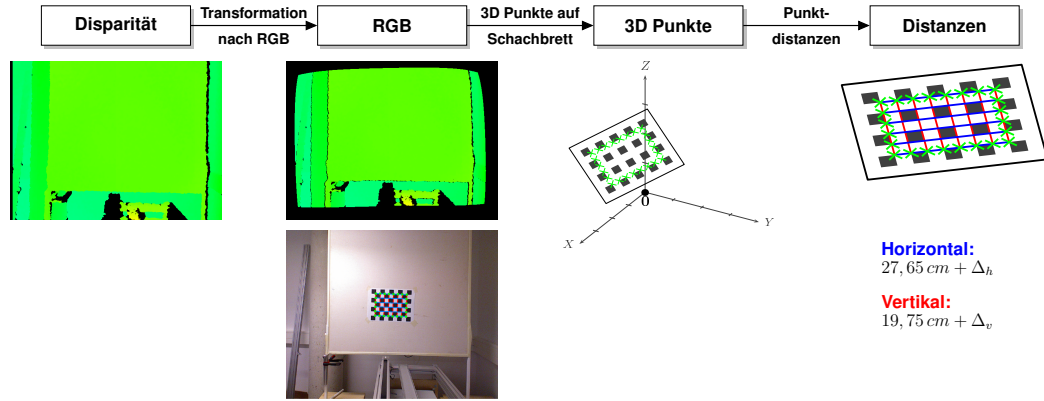


Abbildung 3.57.: Prinzip der 3D-Distanzauswertung

Der Abstand der vertikal angeordneten Punkte, welche in Abbildung 3.57 durch die roten Linien verbunden sind, entspricht im originalen Kalibrierobjekt $19,75 \text{ cm}$. Die Distanz zwischen den durch die blauen Linien aus Abbildung 3.57 verbundenen Punkte beträgt auf dem Schachbrett $27,65 \text{ cm}$. Zur Auswertung in Kapitel 4.2 werden die Fehler $\Delta_v = l_m - 19,75 \text{ cm}$ bzw. $\Delta_h = l_m - 27,65 \text{ cm}$, also die Abweichungen der jeweiligen Messung (Punktdistanzen der vertikalen bzw. horizontalen Punktdistanzen) zur zuvor bekannten Entfernung am Kalibrierobjekt, berechnet. Die für jede aufgenommene Position bestimmten Fehler werden im Unterkapitel *Tiefenmessung 2* von Kapitel 4.2 in einem Boxplot (Kapitel 3.8.1) dargestellt.

3.7.2. Aufbau Datenbank und Aufnahmen der Pferde

Mit dem in Kapitel 3.3 beschriebenen System wurden mehrere Aufnahmen der Tiere aus Kapitel 3.1 in der Versuchsstation, deren Aufbau in Kapitel 3.2 dargestellt ist, aufgezeichnet. Sobald sich die Bilderfassungseinheit aus Kapitel 3.3 synchronisiert hatte, wurde ständig mit Hilfe des Disparitätenbilds eine Maske zur Trennung von Vorder- und Hintergrund erstellt (siehe nächstes Unterkapitel). Die Aufnahme wurde gestartet, sobald in der Maske mehr als 5% der Gesamtzahl ihrer Pixel als Vordergrund detektiert worden sind. Die Beschreibung der Aufnahmen findet sich in den Unterkapiteln *Aufnahme der Daten* bzw. *Variationen in den Aufnahmen*.

Maske zur Trennung von Vorder- und Hintergrund

Zu Beginn wurde ein Hintergrund angelernt, indem über einen kleineren Zeitraum für jeden Pixel im Disparitätenbild jeweils der Wert gespeichert wurde, welcher der größten Entfernung zur Kamera entspricht. Durch den direkten Vergleich des aktuellen Disparitätenbilds mit dem Hintergrund kann eine Maske erzeugt werden, die entsprechend den Vordergrund, welcher sich von dem zuvor aufgezeichneten Hintergrund abhebt, trennt.

Zur genaueren Beschreibung sind hier zwei Gruppen von Aufnahmen zu unterscheiden. Die zuerst aufgezeichneten Daten wurden mit stets geöffneter Wand des V-Ausschnitts (siehe Kapitel 3.2) durchgeführt. Diese Aufnahmen waren meist von Personen begleitet, die die Tiere in die Station geführt haben, welches sich jedoch als sehr zeitaufwändig dargestellt hat. Die ständig geöffnete Wand stellt bei sich frei bewegenden Tieren (ohne durchgehende Beobachtung) eine Verletzungsgefahr für die Tiere dar. Daher wurden alle weiteren Aufnahmen, die den Großteil ausmachen, mit sich öffnender und wieder schließender Wand aufgezeichnet (entsprechend der Beschreibung in Kapitel 3.2). Für die Maske bedeutet dies einen sich verändernden Hintergrund bzw. für den Fall, dass die Wand selbst ebenfalls als Hintergrund angesehen wird, einen Hintergrund, der sich teilweise vor dem Vordergrund befindet. Ohne veränderten Hintergrund (Wand durchgehend unten) werden lediglich solche Pixel als Vordergrund in die Maske aufgenommen, deren Disparitätenwert kleiner dem dazugehörigen Disparitätenwert des Hintergrunds ist. Dies entspricht von den Entfernungswerten solchen Pixeln die sich näher zur Kamera hin befinden, da sich die Disparitätenwerte d , im Fall der Kinect, proportional zur Entfernung verändern (siehe Kapitels 3.3.2). Mit verändertem Hintergrund wird die geschlossene Wand als Hintergrund angenommen. Anschließend wird die absolute Differenz zwischen den Disparitätenwerten des aktuellen Frames und den zuvor aufgezeichneten Hintergrundwerten gebildet. Ist die absolute Differenz kleiner acht, so wird der Pixel als Hintergrund nicht in der Maske markiert. Ist die absolute Differenz größer, so wird der Pixel als Vordergrund angesehen. Dadurch lässt sich eine Veränderung in der Tiefe der aufgezeichneten Szene erkennen. Die absolute Differenz der Disparitätenwerte von acht, entspricht in der Entfernung von $1,5m$, in der sich die Wand zur Kamera befindet, einem Bereich von etwa $10cm$. Der Nachteil dieses Vorgehens ist,

3. Material und Methoden

dass ein sich in diesem Bereich befindlicher Tierkopf in der Maske als Bestandteil der Wand interpretiert wird.

Aufnahme der Daten

Sobald durch die im vorherigen Unterkapitel beschriebene Maske ein Objekt im Vordergrund detektiert wurde (mehr als 5% der Pixel sind Vordergrund), wird eine Aufnahme initiiert. Dazu wird zunächst der zuvor angelernte Hintergrund in einem Container gespeichert. Anschließend werden durchgehend 5 Frames pro Sekunde aufgenommen und ebenfalls im Container abgelegt, bis mindestens 5 Sekunden lang kein Objekt durch die zuvor erwähnte Maske im Vordergrund detektiert wird. Ein Frame umfasst dabei die beiden Bilder der Industriekameras, das Farbbild der Tiefenkamera und das Disparitätenbild der Tiefenkamera. Die Grauwertbilder der Industriekameras werden in einer Auflösung von 1600×1200 gespeichert, die Farbbilder und die Disparitätenbilder der Tiefenkamera in einer Auflösung von 640×480 . Die Bittiefe der Grauwertbilder beträgt 8 Bit, die des Farbbilds pro Kanal ebenfalls 8 Bit (gesamt 24 Bit) und die des Tiefenbilds 16 Bit (wovon 11 Bit verwendet werden). Der Speicherbedarf eines der Grauwertbilder beträgt damit unkomprimiert 1875 KByte, der des Farbbilds 900 KByte und der des Tiefenbilds 600 KByte. Pro Frame werden entsprechend 5,1 MByte gespeichert, so dass für die 5 Frames 25,6 MByte pro Sekunde an Datenmaterial aufgenommen wird. Zudem wird ein Zeitstempel sowie eine Framenummer für jedes Bild in dem Container hinterlegt, wodurch eine möglichst gute Rekonstruktion der Aufnahmen in ihrem zeitlichen Eintreffen ermöglicht wird. Diese zusätzlichen Daten pro Frame sind von ihrem Datenvolumen her zu vernachlässigen.

Parallel zur Aufzeichnung werden die Daten eines jeden Containers direkt auf die Festplatte, in für jeden Besuch einzeln erzeugte Dateien, gespeichert. Hierfür wurde ein eigenes Dateiformat aufgebaut, welches die unterschiedlichen Bilder in ihrer korrekten Reihenfolge mit entsprechender Indizierung, für einen schnelleren Zugriff, abspeichert. Die Bilddaten liegen dabei in einer um 270° rotierten Form zu den in dieser Dissertation gezeigten Darstellungen in der Datei. Dies liegt an der Kameraorientierung, da die Bilder vor dem Speichern nicht rotiert wurden. Um beim Speichern ein möglichst geringes Datenvolumen in Anspruch zu nehmen, ohne Bildqualität einbüßen zu müssen, werden die Farb- und Grauwertbilder als „portable Netzwerkgrafik“ (PNG - *portable network graphics*) [Burger u. Burge 2006] in den Dateien abgelegt. Es handelt sich bei dem PNG Format um ein verlustfreies Kompressionsverfahren, welches alle im Originalbild enthaltenen Informationen erhält. Im Gegensatz dazu steht zum Beispiel das Format „Joint Photographic Experts Group“ (JPEG) [Burger u. Burge 2006], welches eine verlustbehaftete Kompression durchführt, also die Qualität der Bilddaten (wenn auch nur geringfügig) mindert. Die erzeugten Dateien werden entsprechend dem ersten Zeitstempel benannt, so dass eine zeitliche Zuordnung der Dateien ohne weiteren Zugriff möglich ist. Die Kom-

pressionsrate des PNG Formats wird offline, nach Erstellung der Dateien, erhöht, da die Ressourcen des Rechners in der Versuchsstation dies nicht während der Aufnahme zuließen. Dies reduziert das Datenvolumen nochmals ohne dabei Verluste in der Bildqualität zu haben.

Die bei jedem Besuch erzeugten Dateien wurden gesammelt und anschließend manuell einem Tier zugeordnet. Hierzu wurden die einzelnen Dateien in Schritten von höchstens 50 Frames (entspricht 10 Sekunden) durchsucht, um die bei dem Besuch sichtbaren Tiere zuordnen zu können. Waren mehrere Tiere bei einem Besuch sichtbar, so wurden die Dateien gesondert gespeichert, werden jedoch nicht weiter in dieser Dissertation verwendet bzw. wurden in mehrere Dateien aufgeteilt. Der Abstand zur Kamera, in dem ein Tier in einer Datei als sichtbar angenommen wurde, liegt etwa unter 4 m (Schätzung anhand der sichtbaren Umgebung innerhalb der Station auf den 2D Bilddaten). Die Datei, welche einem Besuch eines einzelnen Tieres an der Station zuzuordnen ist, wurde anschließend in dem für das jeweilige Tier angelegten Ordner gesammelt.

Variationen in den Aufnahmen

Frühe Aufnahmen wurden bereits in 2011 durchgeführt (siehe Abbildung 3.58a), wobei der Abstand der Wand in der Versuchsstation zu der Zeit lediglich 94 cm betrug. Erst die Aufnahmen nach 2011 (siehe Abbildung 3.58b) wurden in der im Kapitel 3.2 beschriebenen Station aufgenommen und werden in dieser Dissertation weiter behandelt. Hierzu wurde der gesamte Bereich des Leitsystems von der Kamera weg verschoben, wodurch der Aufnahmebereich auf die in der Skizze 3.3 aus Kapitel 3.2 beschriebenen $1,5\text{ m}$ erweitert wurde. Bei der vor dem Umbau bestehenden Version des Versuchsstalls war der Abstand zwischen V-Ausschnitt und Kamera zu gering, so dass die Tiere bis an die Kamera gelangten. Hierdurch sind vermehrt Frames aufgetreten, in denen keine verwertbaren Tiefendaten aufgezeichnet werden konnten. Außerdem war der Bereich, in dem der Kopf durch die gegebenen Öffnungswinkel der Kameras aufgenommen werden konnte, sehr viel kleiner. Ein Vergleich der Grauwertbilder der unteren Industriekamera ist in Abbildung 3.58 gezeigt. Vor dem Umbau sind Bilder entsprechend der linken Abbildung 3.58a entstanden, danach solche ähnlich Abbildung 3.58b. Zudem hat der vergrößerte Bildausschnitt den Vorteil, dass im unteren Bereich der V-Ausschnitt kurz über dem Trog sichtbar ist. In der Tiefeninformation bedeutet dies, dass dieser Bereich konstant bleibt (Hintergrund) bis das Tier den Kopf nach unten zum Trog bewegt. Die Tiere sind nur unter großer Anstrengung in der Lage, andere Körperteile in diesen Bereich zu bewegen. So kann davon ausgegangen werden, dass es sich bei einer Tiefenänderung in diesem Bereich um einen Tierkopf handelt. Zudem werden im Kapitel 3.7.3 mit Hilfe dieser Bedingung Frames bestimmt, in denen der Kopf den unteren Rand der Tiefenkamera überschreitet. Der Kopf ragt in solchen Aufnahmen so weit aus dem Tiefenbild, dass lediglich für einen Teil des Kopfes Tiefeninformationen vorliegen.

3. Material und Methoden

Vor dem Umbau war ein Bildausschnitt sichtbar, mit dem auf den Boden der Versuchsstation, über den unteren Bereich des V-Ausschnitts hinweg, geschaut wurde. Dieser Bereich war nicht konstant dahin gehend, dass die Hufe im unteren Bereich sichtbar waren, wodurch über eine Tiefenänderung in diesem Bereich nicht auf einen Pferdekopf geschlossen werden kann.

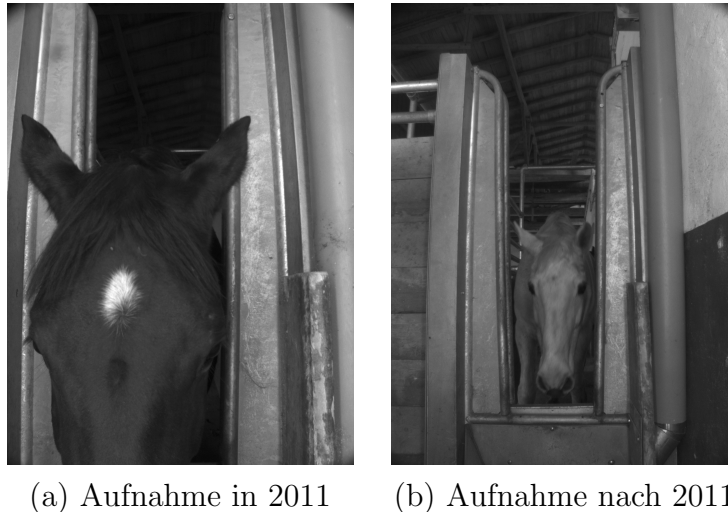


Abbildung 3.58.: Unterschied in den Aufnahmen durch den Umbau nach 2011

Die Standardaufnahmen sahen vor, eine möglichst gute Ausleuchtung der Pferdeköpfe durch die zwei, beidseitig neben der Aufnahmeeinheit angebrachten, Leuchtstoffröhren (siehe Kapitel 3.2) zu erreichen. Dazu waren bei der Mehrzahl der Aufnahmen beide Leuchtstoffröhren eingeschaltet. Um Untersuchungen zur Anfälligkeit von Beleuchtungsänderungen zu ermöglichen, wurden gesondert Besuche aufgezeichnet, bei denen statt beider Leuchtstoffröhren jeweils nur eine angeschaltet war. Hierdurch sollte vermehrt das Auftreten von einseitigen Schattierungen, welche im idealen Fall nicht auftreten sollten, hervorgerufen werden. Diese gesonderten Aufnahmen enthalten vermehrt sich verändernde Helligkeitsunterschiede, welche durch die einseitige Beleuchtung und die Bewegung der Köpfe auf den Bilddaten entstehen.

Zudem konnten für Aufnahmen nach dem 15.01.2013 die RFID Daten aus der Station an den Rechner übertragen werden. Diese wurden gesondert in dem von der Station übertragenen Format gesichert. Bisher waren die Daten vorrangig hilfreich bei der Zuordnung der Tiere, wobei eine manuelle Prüfung trotzdem durchgeführt wurde. Dadurch konnten einige Fehlfunktionen (RFID-Erkennung) in der Station beobachtet werden, bei denen sich ein anderes, als das Tier, welches den Anspruch hatte, in der Station Zugang zum Futter verschafft hat.

3.7. Versuchsdurchführungen

In Tabelle 3.4 sind die in die Datenbank aufgenommenen Besuche aufgeführt. In der ersten und zweiten Spalte findet sich eine Einteilung in die jeweiligen Zeiträume (Zeitraum von, bis), in denen die Besuche aufgezeichnet wurden. Die Kreuze in der dritten bzw. vierten Spalte beschreiben die unterschiedlichen Beleuchtungsarten. Ein Kreuz in der dritten Spalte (Beleuchtung links) bedeutet, dass die linke Leuchtstoffröhre bei einer Draufsicht auf die Aufnahmeeinheit geleuchtet hat und somit das linke Gesichtsfeld der Tiere ausgeleuchtet ist. Entsprechend bedeutet Beleuchtung rechts (vierte Spalte), dass die Leuchtstoffröhre auf der rechten Seite (bei einer Draufsicht) der Aufnahmeeinheit angeschaltet war und die rechte Seite des Kopfes in den Daten ausgeleuchtet ist. Waren beide aktiv (Kreuze in beiden Spalten), so wurde der Tierkopf von beiden Seiten beleuchtet. In den restlichen Spalten ist pro Tier die Anzahl der aufgezeichneten Besuche festgehalten. Die Aufnahmen ab

Tabelle 3.4.: Übersicht der Aufnahmen

Zeitraum		Beleuchtung links	rechts	Anzahl Besuche pro Tier													
von	bis			Anna	Arielle	Baldo	Bara	Blüte	Blütenzauber	Delia	Diadem	Dumbledore	Linus	Romanze	Rosi	Royal Bess	Serenade
10.11. 2011	10.11. 2011	x	x	-	1	-	1	-	-	-	-	-	-	-	-	-	-
13.04. 2012	13.04. 2012	x	x	1	1	-	1	3	1	-	1	-	-	1	5	-	-
24.07. 2012	11.08. 2012	x	x	-	-	4	-	-	-	-	-	-	-	-	13	14	17
10.12. 2012	02.01. 2013	x	x	22	25	-	14	25	21	28	-	13	-	23	23	-	-
04.01. 2013	10.01. 2013	x		23	-	-	20	16	18	18	-	19	-	16	12	-	-
11.01. 2013	27.01. 2013		x	11	18	-	17	20	13	21	-	14	-	12	19	-	-
28.01. 2013	28.02. 2013	x	x	46	39	-	35	39	32	65	-	40	-	34	38	-	-
23.04. 2013	26.04. 2013	x		-	4	-	4	-	2	4	-	2	9	2	4	-	-
26.04. 2013	30.04. 2013		x	-	4	-	4	-	3	2	-	4	4	3	1	-	-
30.04. 2013	06.05. 2013	x	x	-	17	-	-	-	-	13	-	12	4	9	12	-	-
06.05. 2013	07.05. 2013		x	-	5	-	-	-	-	5	-	7	5	3	5	-	-

3. Material und Methoden

dem 10.12.2012 wurden autonom durchgeführt, so dass die Tiere nicht, wie zuvor, in die Station geführt wurden. Trotzdem kam es vor, dass eine Person bei einem Kontrollbesuch in den Aufnahmen zu sehen war.

3.7.3. Vergleichsdaten (Ground Truth)

Die im vorherigen Kapitel 3.7.2 vorgestellte Datenbank ist bereits so strukturiert, dass die Bilddaten mit einer Zuordnung zu einem Tier der Herde aus Kapitel 3.1 gespeichert sind. Die Validierung der vorgestellten Algorithmen bedarf jedoch zusätzlich markierter Daten, die zum einen entsprechende Frames ausweisen, in denen überhaupt ein Tierkopf zu erkennen ist, und zum anderen festlegen, wo sich die Köpfe im Bild befinden. Diese Informationen können dann zum Anlernen (optimieren der Parameter) und zur Kontrolle der Genauigkeit bzw. Robustheit der unterschiedlichen Algorithmen in den einzelnen Schritten der Bildverarbeitungskette dienen.

Die Frames wurden hierzu manuell markiert, da an den Tieren angebrachte Marker, welche das Markieren per Hand überflüssig gemacht hätten, an den Köpfen selber angebracht werden müssten. Dies verdeckt die Tierköpfe jedoch zum Teil, wodurch entsprechende Ergebnisse bei der Erkennung der Gesichter hätten verfälscht werden können. Zur Markierung der Pferdeköpfe wurden lokale Merkmale des Kopfes, nämlich Augen und Nüstern, verwendet. Die Augen und Nüstern wurden hierzu jeweils durch einen einzigen Punkt gekennzeichnet. Das Auffinden der Merkmale auf den Tiefenbildern ist für das menschliche Auge ungewohnt bzw. nur durch die Relationen des gesamten Pferdekopfes möglich. Da dies zu sehr ungenauen Markierungen führt, wurden diese auf den Farbbildern der Tiefenkamera gesetzt. Dies ist vergleichbar mit dem Auffinden der Markerpunkte eines Schachbretts (Kalibrierobjekt), welches im Unterkapitel *Tiefenmessung 2* aus Kapitel 3.7.1 bereits beschrieben wurde.

Die Farbbilder der Tiefenkamera wurden gewählt, da diese Kamera nach dem in Kapitel 3.3 beschriebenen Aufbau den kleinsten Abstand zu der Kamera hat, aus der die Disparitätenbilder stammen. Die räumliche Abweichung kann zwar durch die *Kamerakalibrierung* aus Kapitel 3.3.2 kompensiert werden, wodurch letztlich das Tiefenbild durch entsprechende Transformation mit allen weiteren Kamerabildern in Deckung gebracht werden kann. Trotzdem entstehen durch einen vergrößerten Abstand der Kameras vermehrt Selbstverdeckungen der Tiefeninformation einzelner Objekte in der Szene. Hierdurch sind in den räumlich weiter entfernten Kameras durch eine Transformation größere Bereiche vorhanden, denen keine Tiefeninformation zugeordnet werden kann.

Abbildung 3.59a zeigt das Webinterface, welches zum Setzen der Markierungen aufgebaut wurde. Der Reihe nach können über das Webinterface (aus Sicht des Pferdes) das rechte Auge, das linke Auge, die rechte Nüster und die linke Nüster auf den Köpfen der Tiere markiert werden. Die Farbbilder werden über das Webinterface als Grauwertbilder dargestellt. Dies reduziert zum Einen die Größe der zu transferierenden Daten (schnelleres Laden der Bilder) und erleichtert zum Anderen die bereits gesetzten Marker, welche farbig dargestellt werden, zu erkennen. Das Webin-

terface erlaubt es mehreren Nutzern gleichzeitig die Daten zu markieren. Die auf den Bildern gesetzten Markerpositionen werden in einer zentralen Datenbank abgelegt, welches den Verwaltungsaufwand gering hält. Das Webinterface bietet eine komfortable Möglichkeit, dem Nutzer die entsprechenden Bilder zur Markierung vorzulegen.

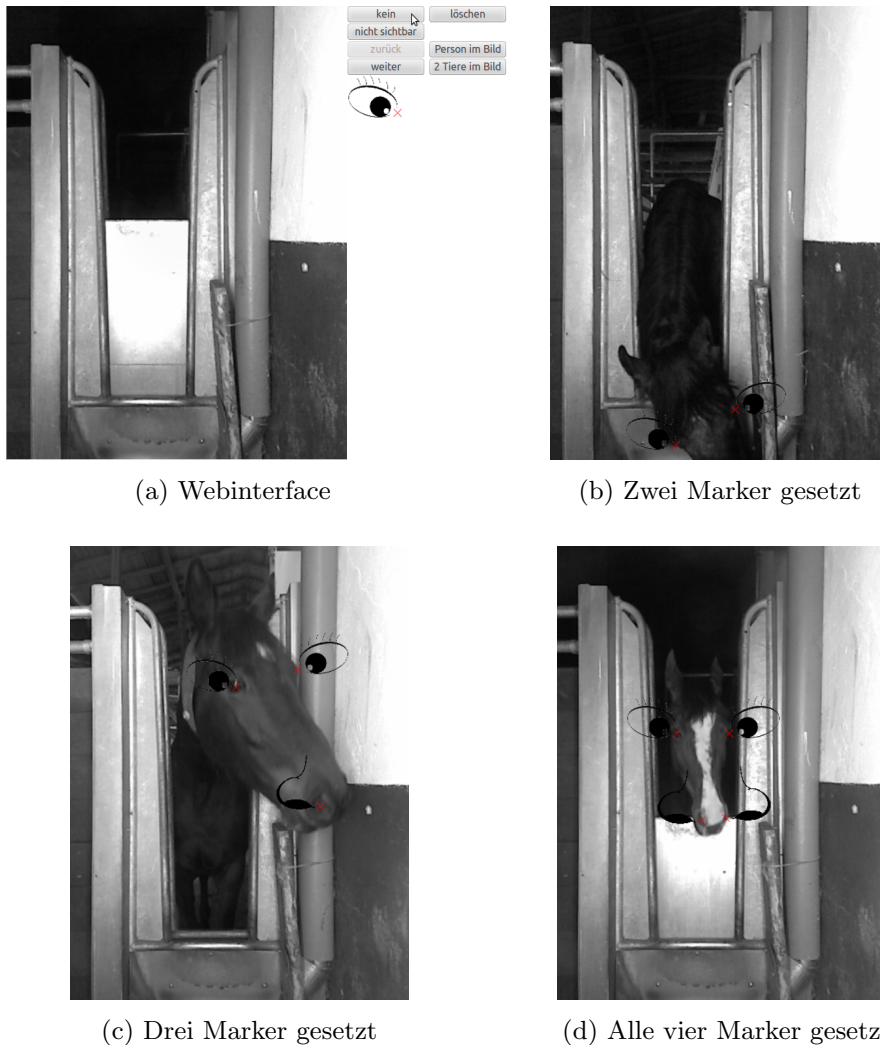


Abbildung 3.59.: Webinterface zum Setzen der Markierungen (Augen und Nüstern)

Die Abbildungen 3.59b-d zeigen unterschiedliche Positionen der Tierköpfe im Webinterface, auf denen eine unterschiedliche Anzahl an Markerpunkten sichtbar ist. Abbildung 3.59b zeigt den im vorherigen Kapitel 3.7.2 beschriebenen Fall, in dem der Kopf nach unten aus dem Farbbild, und übertragen auf das Tiefenbild auch aus diesem, herausragt. Auf diesen Bildern sind nur die beiden Augen auf den Bilddaten erkennbar. In Abbildung 3.59c ist der Pferdekopf hingegen so verdreht, dass es zu

3. Material und Methoden

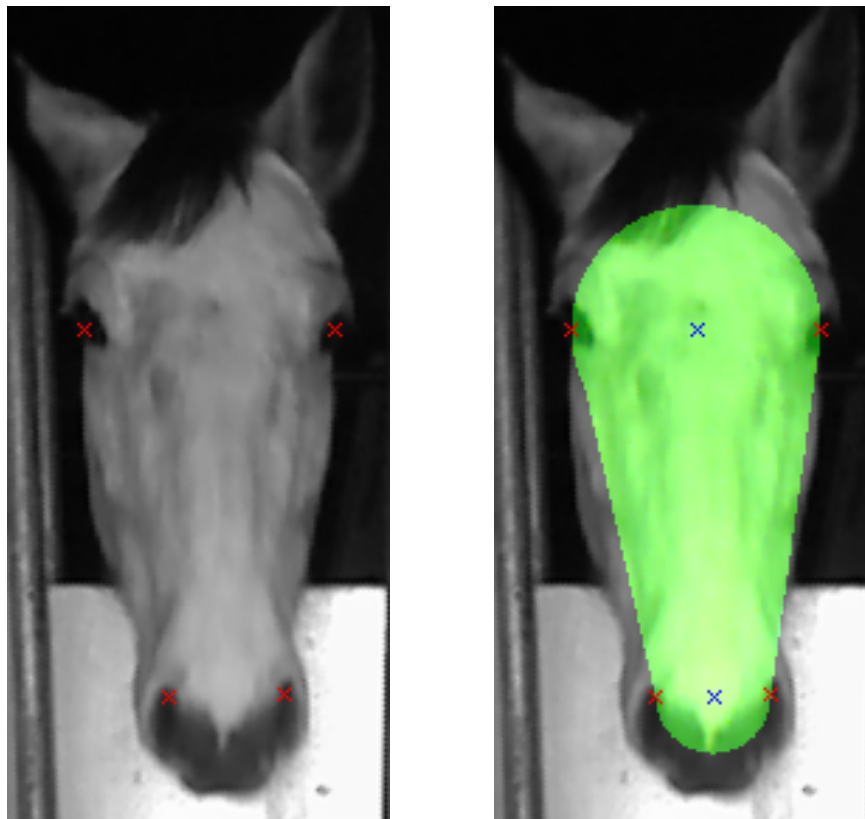
einer Selbstverdeckung der linken Nüster des Tieres kommt, daher sind lediglich drei der insgesamt vier Markierungen gesetzt. Hierbei ist zu erwähnen, dass die Augen aufgrund der Kopfform, welche die Augen im Vergleich zu den Nüstern leicht aus dem Schädel hervorhebt, auch bei größerer Abweichung von der direkten Orientierung zur Kamera sichtbar sind. Die Nüstern hingegen werden deutlich früher vom Schädel des Tieres selbst verdeckt. In Abbildung 3.59d ist der Kopf direkt zur Kamera gerichtet, so dass alle vier Marker gesetzt werden können. Für die in dieser Dissertation beschriebene Erkennung sind ausschließlich solche Bilddaten ähnlich der Abbildung 3.59d von Interesse, in denen beide Augen und Nüstern der Tiere sichtbar sind.

Aus den von Hand gesetzten Markierungen wird eine Maske entsprechend Abbildung 3.60b zur Segmentierung des Kopfes auf den Farbbildern erstellt. Die Masken werden ausschließlich für Bilddaten generiert, in denen alle vier Marker gesetzt wurden. Die Maske wird automatisch generiert, indem die Fläche, welche durch die vier gesetzten Markierungen (rote Kreuze in Abbildung 3.60a) beschränkt ist, extrahiert wird. Zur Maske wird ein oberer Halbkreis, dessen Kreis sein Zentrum im Mittelpunkt zwischen den Augen (oberes blaues Kreuz in Abbildung 3.60b) und einen Durchmesser des Augenabstands hat, hinzugefügt. Ein zweiter Halbkreis wird im Mittelpunkt zwischen den Nüstern (unteres blaues Kreuz in Abbildung 3.60b) mit einem Durchmesser des Abstands der Nüstern auf der Maske markiert. Dies ergibt den in Abbildung 3.60b in grün dargestellten Bereich.

Das Berechnen der Tiefeninformationen der über die Markerpositionen generierten Maske und den darüber markierten Pferdekopf, erfolgt entsprechend dem Vorgehen aus Unterkapitel *Tiefenmessung 2* in Kapitel 3.7.1. Abbildung 3.61 skizziert den Ablauf hierzu nochmals detaillierter. Zunächst werden alle Disparitätenwerte der Tiefenkamera entsprechend der im Anhang C beschriebenen Rekonstruktion in 3D Punkte umgerechnet. Die 3D Punkte sind, wie auch das Disparitätenbild, in der Abbildung farbkodiert dargestellt. Dann werden die 3D Punkte auf das Farbbild projiziert. Das unterste Bild in der Abbildung 3.61 (unter Projektion) zeigt in einem dunklen Grau die Flächen, für die durch die Projektion Tiefenwerte auf dem Farbbild vorhanden sind. Zudem ist die zuvor generierte Maske auf diesem Bild in Weiß dargestellt.

Mit der Maske und der Projektion ist eine Zuordnung der auf dem Farbbild als Teil des Pferdekopfes gekennzeichneten Bereiche und den 3D Daten möglich. Die Punkte, welche durch die Projektion auf das Farbbild innerhalb der Maske liegen, werden nun herausgefiltert (Selektion in Abbildung 3.61). Übrig bleiben die 3D Punkte, welche zum Pferdekopf gehören (3D Pferdekopf in Abbildung 3.61).

Die 3D Punkte eines Pferdekopfes lassen sich nach einer *Kamerakalibrierung* entsprechend auch auf die Kamerabilder der Industriekameras projizieren. Dadurch lässt sich die zuvor manuell erstellte Maske auf die Kamerabilder der Industriekameras übertragen. Hierbei kann es vorkommen, dass die Farbbilder einen leichten zeitlichen



(a) Marker auf Bild (rote Kreuze)

(b) Maske auf Bild mit Markern

Abbildung 3.60.: Maske generiert aus den Markierungen für Augen und Nüstern

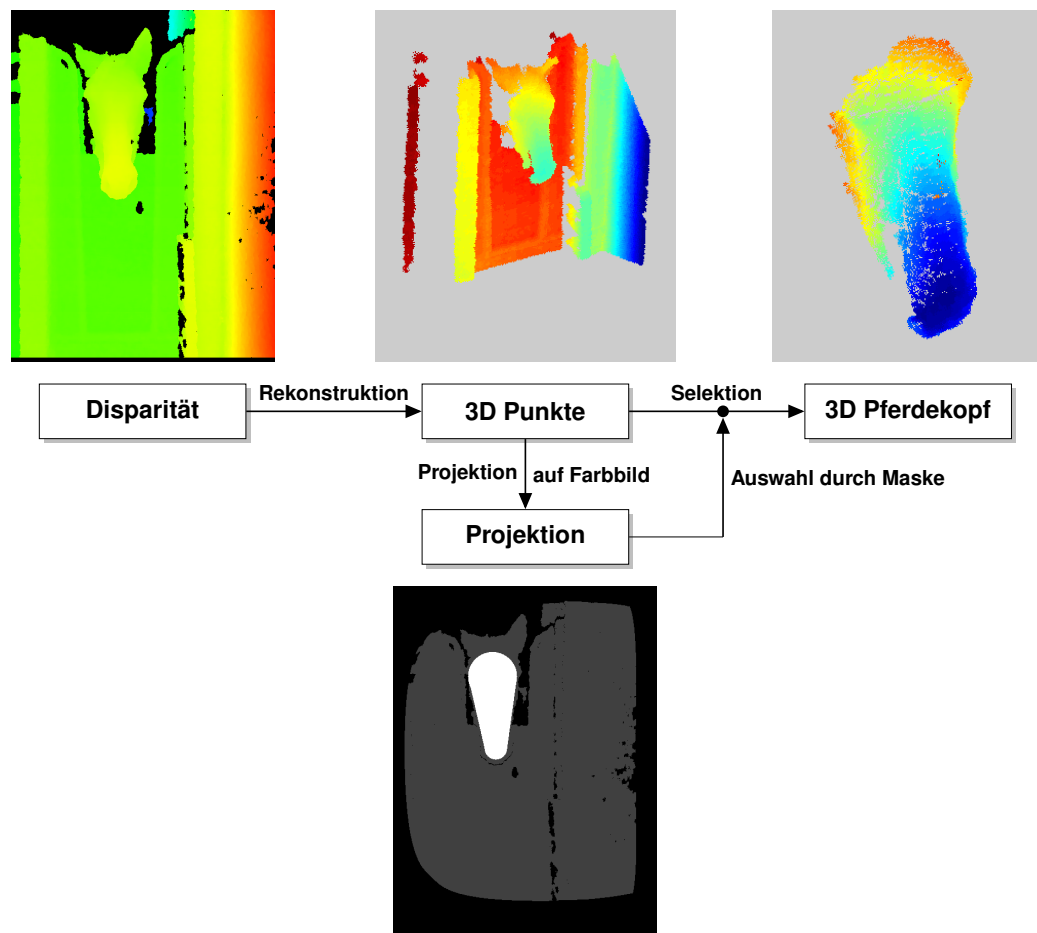


Abbildung 3.61.: Prinzip zur Bestimmung der 3D Daten der markierten Pferdeköpfe

Versatz zu den übrigen Kameraaufnahmen haben, da diese, wie in Kapitel 3.3 beschrieben, nicht mit der Aufnahme des Tiefenbilds per Hardware getriggert werden. Zudem können, durch das direkte Verbinden von Augen und Nüstern, beim Erstellen der Maske, Teile des Hintergrunds mit in die Maske aufgenommen werden. Um sicher zu stellen, dass die extrahierten 3D Daten zu einem Pferdekopf passen und um dessen Orientierung im Raum zu bestimmen, werden Modelle für jedes Tier mit dem im Anhang D beschriebenen Iterative Closest Point-Algorithmus (ICP-Algorithmus) auf die Daten abgebildet. Hierzu wurde zunächst für jedes Tier ein Frame ausgesucht, dessen 3D Daten in ein Modell überführt wurden. Versehentlich über die Maske markierter Hintergrund wurde durch einen manuell festgelegten Schwellwert anhand der Tiefeninformation von jedem Modell entfernt. Eines der Modelle wurde so ausgerichtet, dass es direkt in die Kamera guckt und die Verbindungslinie zwischen den Augen auf der X-Achse liegt. Diese Anordnung eines Modells wird im Folgenden als rektifiziertes Modell bezeichnet. Die übrigen Modelle wurden ebenfalls rektifiziert, indem das manuell ausgerichtete Modell als Template verwendet wurde. Hierzu wurde das Template-Modell auf die Größe des aktuell zu rektifizierende Modell gebracht, da die Köpfe der einzelnen Tiere unterschiedliche Größen haben. Die entsprechende Skalierung der Daten wurde über das Verhältnis der Abstände der Mittelpunkte zwischen den Augen und den Nüstern auf den 3D Daten der Modelle bestimmt. Anschließend wurde das aktuelle Modell durch den ICP-Algorithmus in Übereinstimmung mit dem skalierten Template-Modell gebracht, wodurch dieses ebenfalls in die rektifizierte Position gebracht wurde.

Die für jedes Tier rektifizierten Modelle können mit den extrahierten 3D Daten der markierten Frames durch den ICP-Algorithmus registriert werden. Alle 3D Punkte, welche einen größeren Abstand als 5 cm zu einem der Punkte des eingepassten Modells haben, werden als Hintergrund nicht weiter behandelt. Zusätzlich werden die Markerpunkte in ihre 3D Daten überführt. Dabei werden diese, entsprechend der Abbildung 3.62, um 5% des Abstands zwischen Augen bzw. Nüstern nach innen korrigiert. Dadurch haben die korrigierten Augen (grüne Kreuze in Abbildung 3.62) entsprechend einen Abstand von 90% der zuvor manuell gesetzten Marker (rote Kreuze in Abbildung 3.62). Entsprechendes gilt für die Nüstern. Sollten an den neuen Positionen durch die Projektion der Disparitätenwerte auf das Farbbild keine Tiefeninformation vorhanden sein, wird der entsprechende Marker pixelweise weiter nach innen verschoben, bis für den neuen Pixel eine Tiefeninformation vorhanden ist. Die Korrektur der Markerpositionen wurde vorgenommen, da die Tiefeninformationen vorrangig am Rand der Objekte (Pferdekopf) recht ungenau sein können. Gerade die Augen liegen direkt am Rand des Pferdekopfes, sind jedoch nicht auf einen einzigen Pixel beschränkt, sondern erstrecken sich über eine kleinere Region um die manuell gesetzten Markerpositionen. Die Korrektur der Markierungen soll diese von den Rändern des Pferdekopfes nach innen verschieben, ohne dabei im besten Fall die als Augenregion zu bezeichnende Fläche zu verlassen. Letztlich werden die 3D Koordinaten für die korrigierten Markerpositionen bestimmt.

Die im Modell festgelegten Markierungen (ebenfalls korrigierte Marker) können, ent-

3. Material und Methoden

sprechend der Registrierung durch den ICP-Algorithmus, auf die extrahierten 3D Daten der einzelnen Frames transformiert werden. Durch einen Abgleich der 3D Markierungen des transformierten Modells mit denen des aktuellen Frames werden durch den ICP-Algorithmus fehlgeschlagene Registrierungen ausfindig gemacht. Ist der durchschnittliche Abstand der zu vergleichenden Markierungen kleiner als 20 cm, werden die entsprechenden Frames als Ground Truth (übertragen: Realität, Wirklichkeit) Daten angenommen. Diese Daten dienen als Vergleichsdaten für entsprechende Validierungen. Alle weiteren Frames werden ignoriert.

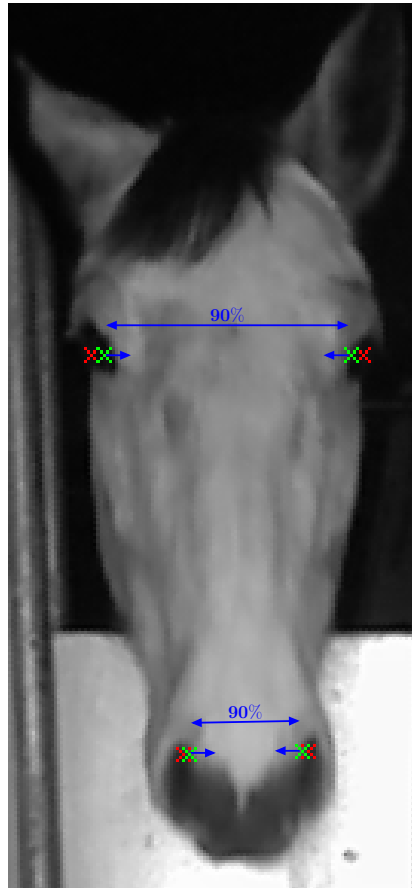


Abbildung 3.62.: Korrektur der Markerpositionen

In diesem Kapitel wurde das Webinterface beschrieben, mit dem die Markerpositionen von Augen und Nüstern von Hand gesetzt wurden. Zudem wird eine Methode vorgestellt, welche dazu dient, die Vergleichsdaten, die aus den 3D Punkten der Pferdeköpfe bestehen, aus den auf dem Farbbild von Hand gesetzten Markierungen zu extrahieren. Die 3D Punkte können nach einer zuvor ausgeführten *Kamerakalibrierung* auf die einzelnen Ebenen der restlichen Kameras projiziert werden. Werden die projizierten Punkte auf den einzelnen Kamerabildern zusammengefasst, lassen sich

Masken für jede Kamera bestimmen. Durch die so bestimmten Masken werden die Pferdeköpfe auf den Bildern vom Rest des Bildes segmentiert.

3.7.4. Orientierung der Pferdeköpfe in den Vergleichsdaten

Durch die Registrierung (siehe Anhang D) der Modelle auf die markierten Daten aus Kapitel 3.7.3, können die Winkel der Abweichung aus der rektifizierten Lage der Modelle bestimmt werden. Der Iterative Closest Point-Algorithmus bestimmt eine Rotationsmatrix \mathbf{R} , sowie einen Translationsvektor \mathbf{t} , um das Modell des jeweiligen Tieres in die 3D Daten des aktuellen Frames zu überführen. Dies ist in Abbildung 3.63 dargestellt. Auf der linken Seite befindet sich das rektifizierte Modell, dessen 3D Punkte farbkodiert dargestellt sind. Mit dem ICP-Algorithmus wird die Transformation (\mathbf{R}, \mathbf{t}) auf die 3D Daten des aktuellen Frames bestimmt. Auf der rechten Seite von Abbildung 3.63 ist das auf die 3D Daten des aktuellen Frames registrierte Modell gezeigt, ebenfalls farbkodiert. Zur Veranschaulichung ist dieses auf das Grauwertbild der Farbkamera projiziert. Zudem sind jeweils die Orientierungen der Koordinatenachsen (rot: Z-Achse, grün: Y-Achse, blau: X-Achse) des rektifizierten und des registrierten Modells dargestellt.

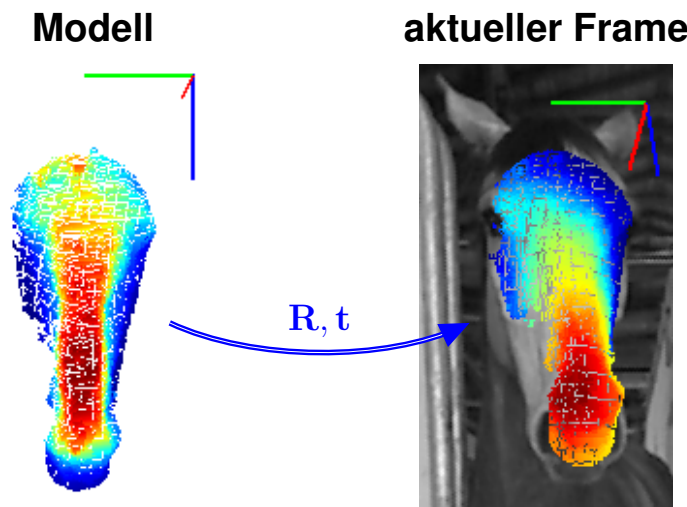


Abbildung 3.63.: Transformation des Modells auf die 3D Daten des aktuellen Frames

Die Z-Achse des transformierten Koordinatensystem entspricht dem Normalenvektor $\mathbf{n}_Z = [N_X \ N_Y \ N_Z]^T$ des registrierten Modells. Der Normalenvektor des rektifizierten Modells entspricht $\hat{\mathbf{n}}_Z = [0 \ 0 \ 1]^T$. Durch die Transformation mit \mathbf{R} , welche durch den ICP-Algorithmus bestimmt wurde, ergibt sich $\mathbf{n}_Z = \mathbf{R} \hat{\mathbf{n}}_Z$. Die Transformation \mathbf{t} hat keinen Einfluss auf Normalenvektoren, da diese lediglich die Orientierung der Daten wiedergeben. Damit entspricht der Normalenvektor \mathbf{n}_Z der letzten Spalte der Rotationsmatrix \mathbf{R} .

3. Material und Methoden

Aus dem Normalenvektor \mathbf{n}_Z lassen sich, durch die Gleichung 3.38 aus Kapitel 3.4.1, die Winkel ψ und θ berechnen:

$$\psi = -\arctan\left(\frac{N_Y}{N_Z}\right) \quad \text{und} \quad \theta = \arctan\left(\frac{N_X}{\sqrt{N_Y^2 + N_Z^2}}\right) \quad (3.108)$$

Für die *Vergleichsdaten* aus Kapitel 3.7.3, deren Abstand zur Kamera unterhalb von $3,0\text{ m}$ liegt, wurden die Winkel bestimmt und in einem Histogramm zusammengestellt. Das Ergebnis ist im Kapitel 4.3 dargestellt.

Zudem wurde eine Einteilung der *Vergleichsdaten* durchgeführt, welche die Frames, die durch auftretende Selbstverdeckung zur Erkennung ungeeignet sind, von denen trennt, die geeignet sind. Hierzu wurden wiederum die Farbbilder der Tiefenkamera begutachtet, da die entsprechende Kamera den geringsten räumlichen Versatz zu der Kamera hat, aus der die Tiefendaten generiert werden. Markiert wurden die Frames, die zuvor mit vier Markern versehen wurden, auf denen also ein Pferdekopf zu erkennen ist. Gekennzeichnet wurde, ob es bei den Bilddaten zu einer Selbstverdeckung kommt, also die Pferdeköpfe eine zu große Orientierungsabweichung von der direkten Sicht in die Kamera haben. In Abbildung 3.64 sind entsprechende Beispiele gezeigt. In der oberen Zeile sind Tiere dargestellt, deren Kopforientierung der direkten Sicht in die Kamera nah genug sind, so dass keine Selbstverdeckung auftritt. Die untere Zeile zeigt solche Bilder, in denen die Köpfe eine Orientierung zur Kamera haben, in der eine Selbstverdeckung auftritt. Entsprechende Bilddaten sind zur Erkennung ungeeignet und werden daher nicht dafür verwendet. In der Abbildung 3.64 ist die Orientierung des Koordinatensystems der durch den ICP-Algorithmus registrierten Modelle durch eine rote, eine grüne und eine blaue Linie gekennzeichnet. Die Ergebnisse dieser Einteilung werden ebenfalls in Kapitel 4.3 behandelt.



Abbildung 3.64.: Beispielbilder ohne und mit Selbstverdeckung

3.7.5. Parameterwahl zur Detektion und Detektionsrate

Die zur Segmentierung der Pferdeköpfe in den Tiefendaten verwendbaren Algorithmen wurden bereits in den Kapiteln 3.4.2 bzw. 3.4.3 vorgestellt. Nachdem entsprechende Flächen in den Bilddaten gefunden wurden, sollen diese mit dem in Kapitel 3.4.4 beschriebenen Verfahren darauf geprüft werden, ob es sich um einen Pferdekopf handelt. Das Vorgehen, Pferdeköpfe innerhalb eines Bildes zu finden, lässt sich damit in zwei aufeinanderfolgende Schritte (I + II) teilen, nämlich das Clustern (Kapitel 3.4.2) bzw. Segmentieren (Kapitel 3.4.3) I und die Detektion der Pferdeköpfe (Kapitel 3.4.4) II.

Qualitätsmaß zur Parameteroptimierung I

Da die einzelnen Algorithmen Parameter besitzen, durch die deren Fähigkeit bei der Durchführung ihrer Aufgabe bestimmt sind, sollen diese möglichst optimal gewählt werden. Um eine entsprechende Wahl für das Clustern (Kapitel 3.4.2) bzw. das Segmentieren (Kapitel 3.4.3), also den ersten Schritt des kompletten Verfahrens zur Detektion, treffen zu können, wird ein Qualitätsmaß eingeführt. Mit dem Qualitätsmaß sollen für einen Satz von Vergleichsdaten, wie sie in Kapitel 3.7.3 beschrieben wurden, die Güte der Parameter bestimmt werden. Genauer werden neben den Disparitätenbildern des Vergleichsdatensatzes, auf denen die Algorithmen arbeiten, Masken benötigt, welche die Pferdeköpfe auf den Daten markieren. Das Generieren der Masken aus den markierten RGB-Bilddaten wurde bereits in Kapitel 3.7.3 beschrieben. Die Masken entsprechen damit den Ground Truth Daten, welche mit den zu bestimmenden Parametern durch die Algorithmen erreicht werden sollen.

\mathbf{M} sei eine Maske, die einer optimalen Segmentierung entspricht. \mathbf{C} ist eine Maske, welche durch den Algorithmus bestimmt wird, dessen Parameter optimiert werden sollen. Die Algorithmen aus Kapitel 3.4.2 bzw. Kapitel 3.4.3 bestimmen mehrere Masken durch die *Regionenmarkierung* bzw. deren Erweiterung, die Region \mathbf{C} wird daher nach der größten Schnittmenge mit \mathbf{M} ausgewählt. Abbildung 3.65 zeigt ein Beispiel, in dem die Maske \mathbf{M} in Weiß und die Maske \mathbf{C} in Rot, sowie deren Schnittmenge ($\mathbf{M} \& \mathbf{C}$) in Grün dargestellt sind. N_M sei die Anzahl der Pixel in der Maske \mathbf{M} (weiß oder grün in Abbildung 3.65). Entsprechend sei N_C die Anzahl der Pixel in der Maske \mathbf{C} (rot oder grün in Abbildung 3.65). $N_{M\&C}$ sei die Anzahl der Pixel, welche sich in der Schnittmenge beider Masken ($\mathbf{M} \& \mathbf{C}$, grün in Abbildung 3.65) befinden.

Das Qualitätsmaß, welches zur Parameteroptimierung Verwendung findet, wird als Jaccard-Koeffizient [Jaccard 1901; Cha 2007] (oder auch Tanimoto-Koeffizient) bezeichnet und ist eine Abwandlung des Inneren Produkts:

$$s_{jac} = \frac{N_{M\&C}}{N_M + N_C - N_{M\&C}} \quad (3.109)$$



Abbildung 3.65.: Beispiel einer Maske durch Segmentierung (**C**: rot + grün) und der gewünschten Maske (**M**: weiß + grün)

Für das Beispiel aus Abbildung 3.65 ist $N_M = 7876$, $N_C = 8199$ und $N_{M\&C} = 6852$, so dass sich $s_{jac} = 0,7429$ ergibt. Stimmen **M** und **C** überein, so ergibt sich für das Qualitätsmaß das Maximum $s_{jac} = 1$. Mit jedem Pixel in **C**, welcher ausserhalb von **M** liegt (roter Bereich in Abbildung 3.65), wird der Nenner in Gleichung 3.109 größer, so dass s_{jac} kleiner wird. Enthält die Schnittmenge aus **M** und **C** weniger Pixel (grüner Bereich in Abbildung 3.65), so wird $N_{M\&C}$ kleiner und damit auch s_{jac} aus Gleichung 3.109. Bei dem Minimum $s_{jac} = 0$ gibt es keine Übereinstimmung von Pixeln in **M** und **C**.

Parameter I

Der Algorithmus aus Kapitel 3.4.2 besitzt 9 zu optimierende Parameter, welche in Tabelle 3.5 zusammengestellt sind.

Tabelle 3.5.: Parameterliste des Cluster-Algorithmus

Parameter	Beschreibung
s_ψ	Schrittweite des Winkels ψ im Hough-Raum (Akkumulator)
s_θ	Schrittweite des Winkels θ im Hough-Raum (Akkumulator)
s_δ	Schrittweite der Distanz δ im Hough-Raum (Akkumulator)
p_ψ	Region um Peak des Winkels ψ im Hough-Raum (Akkumulator)
p_θ	Region um Peak des Winkels θ im Hough-Raum (Akkumulator)
p_δ	Region um Peak der Distanz δ im Hough-Raum (Akkumulator)
T_p	Schwellwert im Hough-Raum (prozentual vom Maximum)
N_{min}	minimale Anzahl Pixel in Region
N_{Rmax}	maximale Anzahl von Regionen

3. Material und Methoden

Die Parameter s_ψ, s_θ und s_δ stellen die Schrittweite der Werte im Akkumulator dar. Die Schrittweite sorgt dafür, dass die Winkel ψ, θ bzw. die Differenz δ quantisiert in die jeweiligen Felder des Akkumulators eingetragen und die Anzahlen der Werte in den jeweiligen Wertebereichen gezählt werden können. Sobald alle Werte im Akkumulator gezählt sind, werden die jeweiligen Maxima gesucht. Da sich meist eine größere Anzahl an Werten um die Peaks (maximale Werte) sammeln, jedoch nicht direkt auf diesen liegen, werden mit den Parametern p_ψ, p_θ und p_δ Bereiche um die Peaks festgelegt. Sobald ein Peak gefunden wurde, werden diese Bereiche um einen Peak herum als Werte dieses Peaks angenommen und beim Auffinden des nächsthöheren Peaks ignoriert. Für die Parameter s_ψ, s_θ, p_ψ und p_θ wurden Grenzen festgelegt, so dass diese nicht kleiner als $0,5^\circ$ und nicht größer als 60° werden. Die Grenzen der Parameter s_δ und p_δ wurden so gelegt, dass die Parameter zwischen 2 cm und 30 cm liegen ($s_\delta, p_\delta \in [2\text{ cm}, 30\text{ cm}]$). Der Parameter T_p wird als prozentualer Schwellwert eingesetzt, welcher, in Abhängigkeit der Anzahl des Peaks mit der höchsten Anzahl N_{max} im Akkumulator, solche Peaks ignoriert, deren Anzahl N unter $T_p N_{max}$ liegen ($N < T_p N_{max}$ mit $T_p \in [0, 1]$). Für die Parameteroptimierung wurde T_p auf maximal $0,97$ beschränkt. N_{min} lässt solche Peaks aus, deren Anzahl N unterhalb von N_{min} liegen ($N < N_{min}$). Die Grenzen für den Parameter N_{min} wurden auf 10 bzw. 2000 festgelegt. Die Maximale Anzahl an Regionen, die durch den Cluster-Algorithmus zurückgegeben werden ist durch N_{Rmax} bestimmt. Bei der Suche wurde N_{Rmax} auf Werte zwischen 1 und 1000 beschränkt.

Der Algorithmus aus Kapitel 3.4.3 besitzt 3 bis 5 Parameter, je nachdem welche der ohnehin berechneten Ebenenparameter zur erweiterten Regionenmarkierung aus Kapitel 3.4.3 berücksichtigt werden.

Tabelle 3.6.: Parameterliste der Segmentierung aus Kapitel 3.4.3

* optionaler Parameter (mindestens einer)

Parameter	Beschreibung
α	Winkelabweichung zur Blickrichtung (Erstellung der Maske)
t_b	Teiler der Blockgröße; Pixelabstands zur Regionenmarkierung
T_ψ^*	Schwellwert zum Distanzmaß von ψ zur Regionenmarkierung
T_θ^*	Schwellwert zum Distanzmaß von θ zur Regionenmarkierung
T_δ^*	Schwellwert zum Distanzmaß von δ zur Regionenmarkierung

α ist die im Kapitel 3.4.3 eingesetzte Winkelabweichung, die zur Generierung der Maske Verwendung findet. Zur Optimierung der Parameter wurde α auf die Werte von 0° bis 90° beschränkt. t_b beschreibt einen Teiler, welcher auf die Blockgröße angewendet wird, die die Nachbarschaft zur Berechnung der Ebenenparameter angibt, woraus sich der Pixelabstand ergibt, der zur erweiterten Regionenmarkierung verwendet wird. Der Teiler durfte bei der Suche des optimalen Parametersatzes Werte zwischen 0 und 32 annehmen, wobei darauf geachtet wurde, dass der berechnete Pixelabstand nicht kleiner 1 (kein Abstand) wurde. Die Schwellwerte T_ψ, T_θ bzw. T_δ

wurden neben der Maske zur erweiterten Regionenmarkierung verwendet und beziehen sich auf die Ebenenparameter ψ, θ bzw. δ . Die Schwellwerte T_ψ und T_θ der Winkel ψ bzw. θ wurden zur Optimierung auf die Werte zwischen 0° und 45° beschränkt. Der Schwellwert T_δ durfte sich zwischen 0 cm und 50 cm bewegen. Durch die optionalen Parameter T_ψ, T_θ und T_δ ergeben sich insgesamt sieben Kombinationen die Parameter im Parametersatz zu berücksichtigen.

Parameteroptimierung I

Zur Optimierung der jeweiligen Parametersätze wurde der Befehl *patternsearch* in MATLAB® [Lewis u. Torczon 1999] mit den bereits angegebenen Schranken zur Parameteroptimierung verwendet. Der mit dem Befehl verwendete Generalized Pattern Search (GPS) Algorithmus gehört zu den Methoden der Direkten Suche (*direct search*) Kolda u. a. [2003]. Diese konvergieren meist nicht besonders schnell, jedoch stellen sie keine Anforderung an die Differenzierbarkeit bzw. die Konvexität der zu minimierenden Funktion und sind deterministisch. Durch die Option *CompleteSearch* (*on*) wird die Kostenfunktion an mehreren Stellen in einer Art „Netz“ über den gesamten Parameterraum ausgewertet, um die beste Position zu bestimmen und an dieser das „Netz“ zu verfeinern. Als Kostenfunktion, welche sich mit den Parametern der Algorithmen aus Kapitel 3.4.3 bzw. Kapitel 3.4.3 verändert, wurde die Gleichung 3.109 für einen Datensatz d

$$c_d = 1 - s_{jac} = 1 - \frac{N_{M\&C}}{N_M + N_C - N_{M\&C}} \quad (3.110)$$

aufgestellt. Die Optimierung wurde jeweils über einen Lerndatensatz von $N_d = 199$ Frames der Vergleichsdaten durchgeführt, wobei die Gesamtkosten c_I

$$c_I = \sum_{d=1}^{N_d} c_d \quad (3.111)$$

der Summe der Einzelkosten c_d aus Gleichung 3.110 der jeweiligen Datensätze entspricht.

Um möglichst gute Lern- bzw. Testdaten zu haben, wurden die zuvor mit Markern versehenen Vergleichsdaten nach dem in Kapitel 4.3 vorgestellten Ergebnis zur Orientierung der Pferdeköpfe ohne Selbstverdeckung kontrolliert. Die 5987 Frames (siehe P' in Tabelle 4.5), in denen die Orientierungswinkel der in den Daten vorhandenen Pferdeköpfe die in Kapitel 4.3 vorgestellte Bedingung erfüllen, wurden in 30 Datensätze unterteilt. Die Zuordnung eines Frames zum Datensatz wurde zufällig gewählt, wobei die Datensätze disjunkt sind, also kein Frame zweifach vorkommt. 15 der insgesamt 30 Datensätze wurden als Lerndaten deklariert, die übrigen 15 als Testdaten. Die Parameteroptimierung wurde einzeln für die 15 Lerndaten über die Kosten der jeweiligen 199 Frames (N_d) nach Gleichung 3.111 mit dem Befehl *pattern-*

3. Material und Methoden

search durchgeführt. Entsprechende Auswertungen zu den gefundenen Parametern sind in Kapitel 4.4 zu finden.

Auswertung der Parameter I

Die Parameter der Algorithmen 3.4.2 bzw. 3.4.3 wurden zunächst nach dem Unterkapitel *Parameteroptimierung I* anhand der 15 Lerndatensätze (je 199 Frames) optimiert. Anschließend werden, neben den jeweiligen Gesamtkosten c_I der Lerndatensätze, die Gesamtkosten \hat{c}_I der entsprechenden Testdatensätze (ebenfalls 15 Datensätze mit jeweils 199 Frames) nach Gleichung 3.111 mit den gefundenen Parametern berechnet. Das Ergebnis ist im Kapitel 4.4.1 in Tabellenform zusammengefasst. Für den Algorithmus aus Kapitel 3.4.2 existiert keine Auswahl optionaler Parameter, daher ergibt sich für den Cluster-Algorithmus eine einzige Tabelle (4.6). Der Algorithmus aus Kapitel 3.4.3 besitzt hingegen eine Auswahl an optionalen Parametern, wie sie in Tabelle 3.6 aufgeführt sind. Insgesamt ergeben sich aus deren Kombination 7 mögliche Parametersätze, daher ist in Kapitel 4.4.1 eine Tabelle (4.7) zur Übersicht der Gesamtkosten c_I und \hat{c}_I angegeben. Im Anhang E.1.1 ist zudem pro Parameterkombination jeweils eine Tabelle (E.1 - E.7) mit den optimierten Parametern zu finden.

Die verwendete Kostenfunktion (Gleichung 3.111) ist unabhängig von den verwendeten Parameterkonstellationen, da das Ergebnis der Algorithmen mit einer zuvor als optimal festgelegten Segmentierung verglichen wird. Die Kostenfunktion wird jeweils auf die gleichen Lern- sowie Testdatensätze angewendet. Dadurch werden nicht nur die einzelnen Parameterkombinationen des Segmentierungs-Algorithmus (nach Kapitel 3.4.3) miteinander vergleichbar, sondern auch die beiden unterschiedlichen Algorithmen aus Kapitel 3.4.2 bzw. Kapitel 3.4.3.

Qualitätsmaß zur Parameteroptimierung II

Die Optimierung der Parameter zur Detektion der Pferdeköpfe aus Kapitel 3.4.4 wird über die in Kapitel 3.8.2 vorgestellte Fläche unter der ROC-Kurve durchgeführt. Diese bietet ein Maß für die Güte der Detektion bzw. eines Klassifikators, welcher eine kontinuierliche Zahl zur Unterscheidung der Klassen zurückgibt. Die Werte der Fläche unter der ROC-Kurve liegen immer zwischen 0 und 1.

Die ROC-Kurve wird mit einem Datensatz erzeugt, dessen Elemente jeweils einer von zwei Klassen zugeordnet sind. Dazu werden die zuvor im Schritt I durch die Algorithmen aus Kapitel 3.4.2 bzw. Kapitel 3.4.3 erzeugten Regionen den zwei Klassen **p** (positiv) und **n** (negativ) zugeordnet. Die Region mit der höchsten Anzahl an Pixeln innerhalb der zuvor im Unterkapitel *Qualitätsmaß zur Parameteroptimierung I* verwendeten Maske, welche einer optimalen Segmentierung nach Kapitel 3.7.3 entspricht, wird als Klasse **p** definiert. Alle übrigen Regionen werden der Klasse **n** zugeordnet. Durch ein Verschieben der 3D Daten der rektifizierten Modelle aus Kapitel 3.7.3 mit üblicher Orientierung (siehe Kapitel 4.3) zwischen 1 m und 3 m und der entsprechenden Projektion auf die Tiefenkamera hat sich ergeben, dass die Größen der projizierten Pferdeköpfe zwischen 1.200 und 30.000 Pixeln liegen. Regionen, deren Pixelanzahl ausserhalb dieser Schranken (1.200 – 30.000 Pixel) liegen, werden entsprechend der Klasse **n** zugeordnet.

Zur Klassifikation wurde der im Kapitel 3.4.4 beschriebene Ähnlichkeitsfaktor ρ aus Gleichung 3.55 verwendet. Der Ähnlichkeitsfaktor ρ bildet dabei die Differenz der Einträge zweier Histogramme $\tilde{H}_S - \tilde{H}_T$ ($\tilde{H}_S \in \tilde{\mathcal{H}}_S$ und $\tilde{H}_T \in \tilde{\mathcal{H}}_T$). Das Histogramm $\tilde{\mathcal{H}}_S$ wird dazu aus den Werten der Ebenenparameter innerhalb der zu untersuchenden Region bestimmt. $\tilde{\mathcal{H}}_T$ ist das Template mit dem verglichen wird und welches aus dem Durchschnitt der Histogramme aller Regionen der Klasse **p** in dem jeweiligen Trainingsdatensatz bestimmt wird.

Parameter II

Das Histogramm-Template $\tilde{\mathcal{H}}_T$ kann als ein Parameter aufgefasst werden, welcher direkt aus dem Trainingsdatensatz bestimmt wird. Die in den Histogrammen, welche zur Auswertung verwendet werden, angewandten Wertebereichsschritte (Bingrößen im Histogramm) sind daher die im Schritt II zu optimierenden Parameter. Zur Auswertung wurden jeweils ein oder zwei der Ebenenparameter in einem Histogramm verglichen, die Bingrößen der Histogramme sind daher in Tabelle 3.7 als optionale Parameter aufgeführt. Der Algorithmus aus Kapitel 3.4.4 erlaubt zwar grundsätzlich den Vergleich von Histogrammen mit drei Dimensionen (alle 3 Parameter), welches jedoch nicht weiter verfolgt wird.

Die Parameter b_ψ und b_θ wurden bei der Optimierung auf die Werte von 0,5° bis 45° beschränkt. Die Suche nach dem optimalen Wert von b_δ wurde auf den Bereich von 0,5 cm bis 15 cm eingeschränkt.

3. Material und Methoden

Tabelle 3.7.: Parameterliste der Detektion von Pferdeköpfen aus Kapitel 3.4.4

* optionale Parameter (ein bis zwei der Parameter)

Parameter	Beschreibung
b_ψ^*	Schrittweite der Werte ψ im Histogramm (Bingröße)
b_θ^*	Schrittweite der Werte θ im Histogramm (Bingröße)
b_δ^*	Schrittweite der Werte δ im Histogramm (Bingröße)

Parameteroptimierung II

Entsprechend der Parameteroptimierung aus Schritt I, werden die optimalen Parametersätze mit dem Befehl *patternsearch* in MATLAB® [Lewis u. Torczon 1999] gesucht. Für jeden der im Unterkapitel *Parameteroptimierung I* beschriebenen 15 Lerndatensätze werden die bereits optimierten Parameter zum Clustern bzw. Segmentieren verwendet. Um dann die unterschiedlichen Kombinationen der Parameter aus Tabelle 3.7 anhand der Kostenfunktion c_{II}

$$c_{II} = 1 - AUC \quad (3.112)$$

zu optimieren. AUC ist dabei die bereits beschriebene Fläche unter der ROC-Kurve, welche entsprechend dem Unterkapitel *Qualitätsmaß zur Parameteroptimierung II* Werte zwischen 0 und 1 annimmt. Da für eine gute Klassifikation der Wert der AUC möglichst groß sein soll, wird eine Minimierung mit der Kostenfunktion c_{II} aus Gleichung 3.112 durchgeführt.

Auswertung der Parameter II und Detektionsrate

Entsprechend der *Auswertung der Parameter I* wurden zunächst die optimierten Parameter nach der *Parameteroptimierung II* für die 15 Lerndatensätze (siehe *Auswertung der Parameter I*) bestimmt. Anschließend werden die Gesamtkosten \hat{c}_{II} nach Gleichung 3.112 für die Testdatensätze berechnet. Die Lern- und Testdatensätze entsprechen damit den Datensätzen, die zuvor zur Optimierung und Auswertung der Parameter in Schritt I (*Parameteroptimierung I*) verwendet wurden. Durch die Kombination der optionalen Parameter aus Tabelle 3.7 ergeben sich 7 unterschiedliche Parametersätze, wobei die Kombination aller drei Kanäle unberücksichtigt bleibt (siehe *Parameter II*), so dass 6 Kombinationen verbleiben.

Die Auswertung in Kapitel 4.4.1 stellt die mit den optimierten Parametern berechneten Gesamtkosten der Lern- c_{II} bzw. Testdatensätze \hat{c}_{II} in Tabellen (4.8 bzw. 4.9) zusammen. Im vorherigen Schritt I wurden 7 mögliche Parameterkombinationen des Segmentierungs-Algorithmus (nach Kapitel 3.4.3) vorgestellt und ausgewertet (Kapitel 4.4.1). In Kombination ergeben sich 42 Parameterkombinationen des Schritt I (7 Parametersätze) und II (6 Parametersätze) für die Kombination der Segmentierung aus Kapitel 3.4.3 und der Detektion aus Kapitel 3.4.4. Aus Gründen der Übersicht beschränkt sich die Auswertung der Detektion des Schritts II in Kapitel 4.4.1 auf die

Parameterkombination des Segmentierungs-Algorithmus (Kapitel 3.4.3) aus Schritt *I*, deren Auswertung in Kapitel 4.4.1 nach der *Auswertung der Parameter I* das beste Ergebnis für den Schritt *I* erlangt. Die Tabellen der verbleibenden Parameterkombinationen aus dem Schritt *I* werden im Anhang E.1.2 aufgeführt. Die optimierten Parameter der Detektion (Kapitel 3.4.4) aus Schritt *II*, deren Gesamtkosten im Kapitel 4.4.1 behandelt werden, sind ebenfalls für die Algorithmen aus Kapitel 3.4.2 und Kapitel 3.4.3 im Anhang E.1.2 in Tabellen aufgeführt.

Die im vorherigen Unterkapitel *Parameteroptimierung II* beschriebenen Kostenfunktionen c_{II} bzw. \hat{c}_{II} verwenden die Fläche unter der ROC-Kurve (AUC). Die AUC gibt nach Kapitel 3.8.2 die Klassifikationseigenschaften eines Klassifikators mit nur einer Zahl wieder. Zur Auswertung der Detektionsraten werden die gesamten ROC-Kurven, welche zuvor zur Bestimmung der AUC in den Kostenfunktionen \hat{c}_{II} der Testdatensätze dienten, in Kapitel 4.4.2 dargestellt.

Die jeweils zwei besten ROC-Kurven der Parameterkombinationen aus Schritt *II* sind für die jeweiligen Algorithmen des Schritt *I* (Kapitel 3.4.2 bzw. Kapitel 3.4.3) in einem Graphen für jeden Testdatensatz zusammengefasst. Zudem werden die zu den ROC-Kurven gehörenden PR-Kurven angegeben, da die Anzahl N der negativen Elemente (Klasse **n**) in den Datensätzen deutlich größer ist als die Anzahl P der positiven Elemente (Klasse **p**). Dies ist jedoch nur für den Vergleich der unterschiedlichen Algorithmen relevant, da die jeweilige Anzahl P bzw. N der Datensätze (N wird durch Schritt *I*, dem Clustern bzw. Segmentieren, bestimmt) sich beim Vergleich der Parametersätze jedes Algorithmus nicht ändert.

3.7.6. Parameter zur Posenbestimmung und Landmarkendetektion

Die Posenbestimmung ist in den Kapiteln 3.5.1 bzw. 3.5.2 beschrieben. Entsprechend der Kapitel ist die Bestimmung der Pose in zwei Schritte (A und B) eingeteilt: A) die *Grobe Posenbestimmung* aus Kapitel 3.5.1 und B) die *Posenverfeinerung* aus Kapitel 3.5.2. Entsprechend der Parameterwahl zur Detektion aus Kapitel 3.7.5 werden die Schritte A und B einzeln optimiert.

Parameter zur Groben Posenbestimmung A

Die zwei zu optimierenden Parameter T_N und T_O , welche bei der Groben Posenbestimmung aus Kapitel 3.5.1 zum Einsatz kommen, wurden bereits im Kapitel 3.5.1 beschrieben. T_N ist der prozentuale Anteil der maximal auftretenden Entfernung $\max(d_{i,M_C})$ zum Masseschwerpunkt \mathbf{M}_C , aus dem sich der Abstand $R_d = T_N \max(d_{i,M_C})$ berechnet. R_d ist der Abstand zum Masseschwerpunkt \mathbf{M}_C , welcher dazu dient, den Bereich der Nüstern zu definieren. Aus dem durch R_d bestimmten Bereich wird der Punkt \mathbf{M}_N und anschließend der Vektor \mathbf{V}_{CN} bestimmt. T_O gibt den prozentualen Anteil der Länge des Vektors \mathbf{V}_{CN} vor, mit dem der Abstand $d_O = T_O \|\mathbf{V}_{CN}\|$ bestimmt wird. d_O dient dazu die Ohren vom detektierten Bereich zu trennen, da diese sich frei vom Kopf positionieren können und damit keinen Einfluss auf die Posenbestimmung des Kopfes haben sollten. Die beiden zu optimierenden Parameter, welche zur Anwendung der Groben Posenbestimmung aus Kapitel 3.5.1 eingesetzt werden, sind in Tabelle 3.8 zusammengestellt.

Tabelle 3.8.: Parameterliste der Groben Posenbestimmung aus Kapitel 3.5.1

Parameter	Beschreibung
T_N	Abstand $R_d = T_N \max(d_{i,M_C})$, definiert den Bereich der Nüstern
T_O	Abstand $d_O = T_O \ \mathbf{V}_{CN}\ $, trennt die Ohren vom Kopf

Zur Optimierung wurde T_N auf die Werte zwischen 30 % und 100 % beschränkt. T_O durfte Werte zwischen 0 % und 100 % annehmen, so dass $d_O \leq \|\mathbf{V}_{CN}\|$ ist.

Parameteroptimierung A

Die Parameter der Tabelle 3.8 beeinflussen die Orientierung \mathbf{R}_I der Groben Pose aus Kapitel 3.5.1. Die Translation \mathbf{t}_I wird hingegen direkt aus der vereinzelter Region (Segmentierung aus Kapitel 3.4.3) bestimmt, so dass die Parameter T_N und T_O keinen Einfluss auf \mathbf{t}_I haben.

Die Orientierung der Vergleichsdaten aus Kapitel 3.7.3 wurde bereits in Kapitel 3.7.4 mit dem *ICP-Algorithmus* (Anhang D) anhand vorgegebener Modelle bestimmt. Mit der Kenntnis des Tieres und damit des 3D Modells lässt sich die Transformation (\mathbf{R}, \mathbf{t}) berechnen, welche das Modell in die Daten der Vergleichsdaten einpasst. Mit dem Verfahren aus Kapitel 3.5.1 soll nun eine Rotationsmatrix \mathbf{R}_I bestimmt

werden, welche durch das Anpassen der Parameter T_N und T_O möglichst geringe Abweichungen zur Rotationsmatrix \mathbf{R} der Transformation aus Kapitel 3.7.4 des jeweiligen Vergleichsdatensatzes hat. Verwendet wurden die 15 Lern- bzw. Testdatensätze aus Kapitel 3.7.5 mit jeweils $N_d = 199$ Frames, auf denen die *Detektion* (Segmentierungs-Algorithmus aus Kapitel 3.4.3 und die Detektion aus Kapitel 3.4.4) mit den Parametern aus der *Parameterwahl* (Kapitel 4.4.3) angewendet wurde. Zur Auswertung wurde je Datensatz nur der Kopfkandidat berücksichtigt, der die größte Deckung mit der jeweiligen Maske aus den *Vergleichsdaten* des Kapitels 3.7.3 aufweist.

In der Arbeit Huynh [2009] werden unterschiedliche Metriken zum Vergleich von Rotationsmatrizen vorgestellt. Diese werden jedoch nicht angewendet, da die Ergebnisse mit der *Parameteroptimierung B* vergleichbar bleiben sollen. Bei der *Parameteroptimierung B* wird hingegen neben der Rotation $\mathbf{R}_{\mathbf{II}+\mathbf{III}}$ auch die Translation $\mathbf{t}_{\mathbf{II}+\mathbf{III}}$ der Transformation berücksichtigt, welche zur Optimierung mit einem Kostenwert beschrieben werden sollen. Zwischen den Winkeldifferenzen der Rotationsmatrizen \mathbf{R} und $\mathbf{R}_{\mathbf{II}+\mathbf{III}}$, die nach Huynh [2009] als Metrik verwendbar sind, und dem Abstand der Translationen \mathbf{t} und $\mathbf{t}_{\mathbf{II}+\mathbf{III}}$ wäre eine Gewichtung einzuführen. Da eine solche Gewichtung Einfluss auf die Optimierung hat und somit schwer festzulegen ist, werden die 3D Daten \mathbf{X}_K , die erst zur Posenbestimmung in Kapitel 3.5.2 bestimmt werden, mit den Transformationen $(\mathbf{R}_I, \mathbf{t})$ und (\mathbf{R}, \mathbf{t}) korrigiert

$$\begin{aligned}\mathbf{X}_A &= \mathbf{R}_I^T (\mathbf{X}_K - \mathbf{t}) \\ \tilde{\mathbf{X}}_A &= \mathbf{R}^T (\mathbf{X}_K - \mathbf{t}) .\end{aligned}\tag{3.113}$$

In Gleichung 3.113 wurde nicht \mathbf{t}_I zur Berechnung von \mathbf{X}_A verwendet, sondern die durch den *ICP-Algorithmus* (Anhang D) bestimmte Translation \mathbf{t} . Der Vergleich der gesuchten Transformation $(\mathbf{R}_I, \mathbf{t})$ mit der Transformation (\mathbf{R}, \mathbf{t}) der Vergleichsdaten nach Kapitel 3.7.4 wird durch die Euklidische Distanz d_A (basierend auf der Euklidischen Norm [Bronštejn u. a. 2012, S. 283]) zwischen den Punkten $\mathbf{X}_A = [X_A \ Y_A \ Z_A]^T$ und $\tilde{\mathbf{X}}_A = [\tilde{X}_A \ \tilde{Y}_A \ \tilde{Z}_A]^T$ bestimmt

$$d_A = \|\mathbf{X}_A - \tilde{\mathbf{X}}_A\| = \sqrt{(X_A - \tilde{X}_A)^2 + (Y_A - \tilde{Y}_A)^2 + (Z_A - \tilde{Z}_A)^2} .\tag{3.114}$$

Damit die Anzahl M der Punkte \mathbf{X}_A bzw. $\tilde{\mathbf{X}}_A$ keinen Einfluss auf die Kostenfunktion hat, wird das arithmetische Mittel der Distanzen $d_{A,m}$ gebildet

$$\overline{d_A} = \frac{1}{M} \sum_{m=1}^M d_{A,m} ,\tag{3.115}$$

wobei durch m die jeweilige Punktdistanz der einzelnen Punkte \mathbf{X}_A bzw. $\tilde{\mathbf{X}}_A$ (gesamt M) gekennzeichnet ist. Die Gesamtzahl der M Punkte entspricht der Anzahl der markierten Pixel in der Maske der Abbildung 3.5.2a (roter und weißer Bereich)

3. Material und Methoden

aus Kapitel 3.5.2.

Zur Optimierung der Parameter aus Tabelle 3.8 wird die Kostenfunktion als die Summe der Distanzen \overline{d}_A aus Gleichung 3.115

$$c_A = \sum_{n=1}^{N_d} \overline{d}_{A,n} \quad (3.116)$$

über die Frames n ($n \in [1, N_d]$ mit $N_d = 199$) des jeweiligen Lerndatensatzes aufgestellt. Die Distanz \overline{d}_A wurde zur Optimierung auf eine maximale Länge von $0,2m$ beschränkt. Anschließend werden die Parameter T_N und T_O mit dem Generalized Pattern Search (GPS) Algorithmus [Kolda u. a. 2003], der bereits in Kapitel 3.7.5 verwendet wurde, unter Verwendung der Kostenfunktion aus Gleichung 3.116 optimiert.

Ausnahmefälle, welche die Kostenfunktion aus Gleichung 3.116 nicht beeinflussen sollen, sind keine bzw. eine unrealistische Detektion durch die Kombination des Segmentierungs-Algorithmus aus Kapitel 3.4.3 und der Detektion aus Kapitel 3.4.4. Für diese Fälle bleibt die Kostenfunktion durch den jeweiligen Datensatz unverändert ($\overline{d}_A = 0$ in Gleichung 3.115, betrifft $1 - 5\%$ der Daten eines Lerndatensatzes). Keine Detektion wird angenommen, sollte durch die Detektion keine Deckung mit den Vergleichsdaten vorhanden sein oder sollte der Ähnlichkeitsfaktor ρ aus Gleichung 3.55 (Kapitel 3.4.4) oberhalb von $0,3$ liegen (optimistische Wahl vgl. T_{ROC} aus Tabelle 4.10 in Kapitel 4.4.2). Die unrealistische Detektion wird durch eine Approximation der detektierten Fläche des Kopfkandidaten bestimmt. Die Approximation wird aus der durchschnittlichen Disparität \overline{d} der Disparitätenwerte d innerhalb des detektierten Bereichs (Maske \mathbf{M} weiß dargestellt in der Abbildung 3.34a aus Kapitel 3.5.2) aus Kapitel 3.4.3 berechnet. Ist die Anzahl M der Pixel des in Kapitel 3.5.2 dilatierten Bereichs (Maske $\mathbf{M} \oplus \mathbf{S}$ weiß und rot dargestellt in der Abbildung 3.34a aus Kapitel 3.5.2) kleiner, $M < 28000 - 28 \overline{d}$, oder größer, $M > 115000 - 115,9 \overline{d}$, so wird der Kopfkandidat als unrealistisch angenommen.

Wird der Kopfkandidat durch die Ungleichung $\|\mathbf{V}_{\mathbf{CN}}\| = \|\mathbf{M}_{\mathbf{N}} - \mathbf{M}_{\mathbf{C}}\| < R_d$ aus Kapitel 3.5.1 als Pferdekopf disqualifiziert, so wird der Kostenfunktion die „Strafdistanz“ $\overline{d}_A = 0,2m$ (maximale Länge von \overline{d}_A in Gleichung 3.116) für den jeweiligen Datensatz hinzugefügt.

Auswertung der Parameter A

Die Parameter des Algorithmus *Grobe Posenbestimmung* aus Kapitel 3.5.1 wurden zunächst nach dem Unterkapitel *Parameteroptimierung A* anhand der 15 Lerndatensätze (je 199 Frames) optimiert. Anschließend werden, neben den jeweiligen Gesamtkosten c_A der Lerndatensätze, die Gesamtkosten \hat{c}_A der entsprechenden Testdatensätze (ebenfalls 15 Datensätze mit jeweils 199 Frames) nach Gleichung 3.116

mit den gefundenen Parametern berechnet. Die Ergebnisse sind im Kapitel 4.5.1 in Tabellenform zusammengefasst.

Parameter zur Posenverfeinerung B

In Kapitel 3.5.2 sind vier Parameter der *Posenverfeinerung* zuzuordnen und ein Parameter der *Landmarkendetektion*. Zudem werden zwei weitere Parameter vorgestellt. Insgesamt ergeben sich 7 zu optimierende Parameter, die im Anschluss in Tabelle 3.9 zusammengefasst werden.

Zwei Parameter der *Posenverfeinerung* legen Abstand δ_S und Abmessung Δ_S der Schichten \mathcal{S} entlang der X-Achse fest. Die Schichten \mathcal{S} dienen dazu, die Medianebene Π_S des Pferdekopfes bestimmen zu können. Zwei weitere Parameter sind die Breite Δ_W aus Gleichung 3.63 und die Länge Δ_L aus Gleichung 3.65 des Streifens auf dem Nasenrücken, aus dem die Ebene Π_N in Kapitel 3.5.2 bestimmt wird.

Der Parameter Δ_A legt den Bereich um die Augen $\tilde{\mathcal{A}}$ zur *Landmarkendetektion* in Kapitel 3.5.2 fest.

Zusätzlich wird ein Parameter T_Z verwendet, welcher den maximalen Abstand der durch die *Grobe Posenbestimmung* korrigierten Punkte \mathbf{X}_A (siehe Gleichung 3.113) zum Ursprung entlang der Z-Achse vorgibt. Durch den Parameter T_Z werden die Punkte \mathbf{X}_A mit zu großem Abstand als Ausreißer behandelt und bleiben bei der Posenkorrektur unberücksichtigt.

Zur Optimierung wird ein weiterer Parameter δ_{off} im nächsten Unterkapitel (*Parameteroptimierung B*) eingeführt, welcher nicht für das eigentliche Verfahren aus Kapitel 3.5.2 gebraucht wird. Der Parameter δ_{off} soll die Fehler, welche durch einen leichten Versatz der Augen aus den *Vergleichsdaten* (Kapitel 3.7.3) zu den im Kapitel 3.5.2 beschriebenen Landmarken auf der X-Achse entstehen, während der Optimierung ausgleichen. Der Versatz tritt auf, da die Landmarken Punkte beschreiben, die leicht oberhalb der Augen liegen.

Tabelle 3.9.: Parameterliste der Posenverfeinerung aus Kapitel 3.5.2

Parameter	Beschreibung
δ_S	Abstand der Schichten \mathcal{S} (Berechnung von Π_S)
Δ_S	Abmessung der Schichten \mathcal{S} (Berechnung von Π_S)
Δ_W	Breite des Streifens auf dem Nasenrücken (Π_N)
Δ_L	Länge des Streifens auf dem Nasenrücken (Π_N)
Δ_A	Legt Bereich $\tilde{\mathcal{A}}$ der Augen fest
T_Z	Eliminiert Ausreißer der Punkte \mathbf{X}_A (Abstand Z-Achse)
δ_{off}	Korrektur des Abstands der Landmarken von den Augen

Tabelle 3.10 stellt die obere und untere Schranken der Parameter aus Tabelle 3.10 zusammen, welche die Wertebereiche der Parameter zur Optimierung vorgeben.

3. Material und Methoden

Tabelle 3.10.: Beschränkung der Parameter zur Posenverfeinerung aus Kapitel 3.5.2

Parameter	untere Schranke	obere Schranke
δ_S	0, 1 cm	5 cm
Δ_S	0, 1 cm	25 cm
Δ_W	0, 5 cm	15 cm
Δ_L	5 cm	50 cm
Δ_A	0, 5 cm	50 cm
T_Z	5 cm	20 cm
δ_{off}	0 cm	20 cm

Parameteroptimierung B

Die Optimierung der Parameter aus Tabelle 3.9 erfolgt ähnlich der *Parameteroptimierung A*. Die zu optimierenden Parameter beeinflussen die Orientierung $\mathbf{R}_{\text{II+III}}$ und die Translation $\mathbf{t}_{\text{II+III}}$ der *Posenverfeinerung* aus Kapitel 3.5.2. Die als optimal angenommen Transformation (\mathbf{R}, \mathbf{t}) wurden aus den Vergleichsdaten (Kapitel 3.7.3) extrahiert und entsprechen den zur *Parameteroptimierung A* verwendeten.

Im Unterkapitel *Parameteroptimierung A* wurde bereits darauf eingegangen, dass nicht eine Winkeldifferenz nach Huynh [2009], sondern die Euklidische Distanz als Metrik eingesetzt wird (siehe Gleichung 3.114). Zunächst wird die als optimal angenommene Transformation (\mathbf{R}, \mathbf{t}) aus den *Vergleichsdaten* (Kapitel 3.7.3) auf die 3D Daten \mathbf{X}_K angewendet

$$\tilde{\mathbf{X}}_B = \mathbf{R}^T(\mathbf{X}_K - \mathbf{t}) . \quad (3.117)$$

Dies entspricht der Gleichung 3.113 aus Unterkapitel *Parameteroptimierung A*, so dass die Punkte $\tilde{\mathbf{X}}_B$ und $\tilde{\mathbf{X}}_A$ identisch sind

$$\tilde{\mathbf{X}}_B = \tilde{\mathbf{X}}_A . \quad (3.118)$$

Die Transformation $(\mathbf{R}_{\text{II+III}}, \mathbf{t}_{\text{II+III}})$, welche über die Parameter aus Tabelle 3.9 optimiert werden soll, wird auf die bereits durch die *Grobe Posenbestimmung* aus Kapitel 3.5.1 korrigierten Daten \mathbf{X}_A (entsprechend Gleichung 3.113) angewendet

$$\mathbf{X}_B = \mathbf{R}_{\text{II+III}}^T(\mathbf{X}_A - \mathbf{t}_{\text{II+III}}) . \quad (3.119)$$

Die korrigierten Daten \mathbf{X}_A entsprechen den korrigierten 3D Daten zur *Posenverfeinerung* \mathbf{X}_{KI} aus Kapitel 3.5.2. Die Euklidische Distanz nach Gleichung 3.114 wird leicht verändert

$$\begin{aligned} d_B &= \left\| \mathbf{X}_B - \begin{bmatrix} \delta_{off} \\ 0 \\ 0 \end{bmatrix} - \tilde{\mathbf{X}}_B \right\| \\ &= \sqrt{(X_B - \tilde{X}_B - \delta_{off})^2 + (Y_B - \tilde{Y}_B)^2 + (Z_B - \tilde{Z}_B)^2} . \end{aligned} \quad (3.120)$$

so dass mit δ_{off} ein für alle Lerndatensätze fester Versatz der Landmarken zu den Augen, wie im vorherigen Unterkapitel erläutert, in X-Richtung berücksichtigt wird.

Aus den Distanzen d_B der einzelnen Punkte \mathbf{X}_B und $\tilde{\mathbf{X}}_B$ (Gesamtzahl M) wird, entsprechend der Gleichung 3.115 aus Unterkapitel *Parameteroptimierung A*, das arithmetische Mittel gebildet

$$\overline{d_B} = \frac{1}{M} \sum_{m=1}^M d_{B,m} . \quad (3.121)$$

m gibt die jeweilige Punktdistanz der Punkte \mathbf{X}_B bzw. $\tilde{\mathbf{X}}_B$ (gesamt M) an.

Zur Optimierung der Parameter aus Tabelle 3.9 wird die Kostenfunktion als die Summe der Distanzen $\overline{d_B}$ aus Gleichung 3.121

$$c_B = \sum_{n=1}^{N_d} \overline{d_{B,n}} \quad (3.122)$$

über die Frames n ($n \in [1, N_d]$ mit $N_d = 199$) des jeweiligen Lerndatensatzes aufgestellt. Die Distanz $\overline{d_B}$ wurde zur Optimierung auf eine maximale Länge von $0,2\text{ m}$ beschränkt. Anschließend werden die Parameter aus Tabelle 3.9 mit dem Generalized Pattern Search (GPS) Algorithmus [Kolda u. a. 2003], wie zuvor zur *Parameteroptimierung A*, unter Verwendung der Kostenfunktion aus Gleichung 3.122 optimiert.

Die im Unterkapitel *Parameteroptimierung A* beschriebenen Ausnahmefälle bleiben in der Kostenfunktion der Gleichung 3.122, wie zuvor im Unterkapitel *Parameteroptimierung A*, unberücksichtigt ($\overline{d_{B,n}} = 0$). Zusätzlich wird die Medianebene Π_S nur berechnet, solange mehr als 5 Kreiszentren \mathbf{K}'_C nach Kapitel 3.5.2 detektiert wurden. Sind weniger Kreiszentren vorhanden, so wird keine Posenverfeinerung (Kapitel 3.5.2) durchgeführt, und die Pose wird entsprechend der *Groben Posenbestimmung* aus Kapitel 3.5.1 angenommen. Entsprechendes gilt, sollte die *Landmarkendetektion* keine oder nur eine der beiden Landmarken auffinden. Dann wird der Schritt zur *Finalen Posenverfeinerung mit Landmarken* aus Kapitel 3.5.2 übersprungen.

Wird der Pferdekopf entsprechend der im Unterkapitel *Parameteroptimierung A* beschriebenen Ungleichung disqualifiziert, so wird der Kostenfunktion die „Strafdistanz“ $\overline{d_B} = 0,2\text{ m}$ (maximale Länge von $\overline{d_B}$ in Gleichung 3.122) für den jeweiligen Datensatz hinzugefügt.

Auswertung der Parameter B

Die Parameter des Algorithmus zur *Posenverfeinerung und Landmarkendetektion* aus Kapitel 3.5.2 wurden zunächst nach dem Unterkapitel *Parameteroptimierung B* anhand der 15 Lerndatensätze (je 199 Frames) optimiert. Anschließend werden,

3. Material und Methoden

neben den jeweiligen Gesamtkosten c_B der Lerndatensätze, die Gesamtkosten \hat{c}_B der entsprechenden Testdatensätze (ebenfalls 15 Datensätze mit jeweils 199 Frames) nach Gleichung 3.122 mit den gefundenen Parametern berechnet. Die Ergebnisse sind im Kapitel 4.5.1 in Tabellenform zusammengefasst.

Parameter zur Normalisierung der 2D Daten

Die Kameramatrix \mathbf{K}_V aus Kapitel 3.5.3 wird neben der bereits beschriebenen Pose (Rotation und Translation), mit der die extrinsische Kameramatrix bestimmt wird, dazu genutzt ein virtuelles Kamerabild zu generieren. Im Unterkapitel *Virtuelle Kamera* wird beschrieben, wie für jeden Frame die entsprechende Kameramatrix \mathbf{K}_V bestimmt wird. Zur Bestimmung von \mathbf{K}_V wird ein zuvor festgelegter Pixelabstand Δ_{px} der Landmarken auf dem zu erzeugenden virtuellen Kamerabild und der tatsächliche Abstand Δ_P (im \mathbb{R}^3) der Landmarken aus Kapitel 3.5.2 genutzt. Durch eine ungenaue Bestimmung der Landmarken, und damit deren Abstand Δ_P , kann jedoch die Größe der Abbildung des Pferdekopfes eines Tieres schwanken, wodurch die Identifikation beeinträchtigt wird. Daher wird die Kameramatrix \mathbf{K}_V jeweils für alle Lerndatensätze aus Kapitel 3.7.5 bestimmt, um anschließend eine einheitliche intrinsische Kameraabbildung (die extrinsische Kameramatrix bleibt durch die Pose beeinflusst) für die weiteren Auswertungen, durch die Mittelung aller Kameramatrizen \mathbf{K}_V , zu berechnen. Dadurch erzeugen unterschiedlich große Pferdeköpfe bei der Abbildung auch unterschiedlich große normalisierte Grauwertbilder bzw. bei gleichbleibend großen Köpfen ist die Abbildungsgröße identisch.

Das Ergebnis, die Kameramatrix \mathbf{K}_V , ist im Kapitel 4.5.3 zu finden.

3.7.7. Auswahl der Daten und Vorgehen zur Identifikation

In den Frames der Aufnahmen aus Tabelle 4.1 (Kapitel 4.1) wurden zunächst mit dem Verfahren aus Kapitel 3.4.3 mögliche Kopfkandidaten in den Tiefendaten auffindig gemacht. Anschließend wurde mit der in Kapitel 3.4.4 beschriebenen Detektion geprüft, ob es sich bei den Kopfkandidaten tatsächlich um Pferdeköpfe handelt. Zum Einsatz kamen dabei die Parameter der *Parameterwahl* aus Kapitel 4.4.3. Der Schwellwert T_{ROC} (siehe Tabellen 4.10-4.13 in Kapitel 4.4.2), über den mit dem Ähnlichkeitsfaktor ρ aus Gleichung 3.55 in Kapitel 3.4.4 die Klassifikation vorgenommen wird, ist auf $T_{ROC} = 0,35$ festgelegt. T_{ROC} liegt in den Tabellen 4.10 und 4.11 mit der zum Einsatz gekommenen Segmentierung aus Kapitel 3.4.3 in allen Fällen unterhalb von $T_{ROC} = 0,26$, so dass die Wahl mit $T_{ROC} = 0,35$ nicht sehr stringent ist. Mit dieser „weichen“ Klassifikation werden mehr Frames als Pferdeköpfe klassifiziert, so dass wenige Frames, die tatsächlich einen Pferdekopf im Bild haben und „falsch“ klassifiziert werden (FN - Falsch-Negativ siehe Abbildung 3.70 in Kapitel 3.8.2), verworfen werden.

Zur Normalisierung der Grauwertbilder aus Kapitel 3.5.3, wobei eine virtuelle Ansicht auf die Pferdeköpfe generiert wird, wurde zunächst die Pose nach Kapitel 3.5.1 und 3.5.2 der zuvor detektierten Pferdeköpfe (Kapitel 3.4) bestimmt. Die zur Posenbestimmung und Normalisierung eingesetzten Parameter stammen aus Kapitel 4.5.2 bzw. 4.5.3.

Symmetrieprüfung

Zusätzlich zur Detektion aus Kapitel 3.4.4 wurden die Kopfkandidaten auf Symmetrie geprüft. Zur Symmetrieprüfung werden die korrigierten 3D Daten \mathbf{X}_V aus dem Koordinatensystem der virtuellen Kamera (siehe Kapitel 3.5.3) mit den an der X-Z-Ebene gespiegelten Punkten $\mathbf{X}_{V\star}$ aus Kapitel 3.5.3 verglichen. Die originalen \mathbf{X}_V , als auch die gespiegelten Punkte $\mathbf{X}_{V\star}$ werden mit der Kameramatrix \mathbf{K}_V auf die *Virtuelle Kamera* (siehe Kapitel 3.5.3) projiziert. Anschließend wird, bei vorgegebener Auflösung des virtuellen Bildes (siehe Kapitel 4.5.3), überprüft wieviele der Punkte aus \mathbf{X}_V bzw. $\mathbf{X}_{V\star}$ sich mit den jeweils gespiegelten Punkten $\mathbf{X}_{V\star}$ bzw. \mathbf{X}_V auf dem Kamerabild in einem Pixel decken. Der prozentuale Anteil der sich deckenden Pixel sollte größer oder gleich 70 % sein, wobei zusätzlich das arithmetische Mittel der Euklidischen Distanzen entsprechend der Gleichung 3.115 (Kapitel 3.7.6) der sich deckenden Punkte \mathbf{X}_V und $\mathbf{X}_{V\star}$ unterhalb von 1 cm liegen sollte. Decken sich weniger als 70 % der gespiegelten Punkte $\mathbf{X}_{V\star}$ mit den originalen Punkten \mathbf{X}_V des Koordinatensystems der virtuellen Kamera oder ist deren Distanz im Durchschnitt (arithmetisches Mittel) größer als 1 cm, so wurde der entsprechende Frame zur weiteren Verarbeitung ignoriert.

Die Anzahl der Frames, welche nach der *Detektion* (Kapitel 3.4) und der *Symmetrieprüfung* als Kopfkandidaten angenommen werden, sind in Tabelle 4.22 im Kapitel 4.6.1 aufgeführt.

Gruppieren der Daten

Die nach der *Symmetrieprüfung* verbleibenden Daten enthalten neben den eigentlichen Pferdeköpfen weiteres Bildmaterial, welches nach der Normalisierung der Grauwertbilder aus Kapitel 3.5.3 nicht zur Identifikation geeignet ist. Daher wurden alle 89.662 Frames der Tabelle 4.22 (Kapitel 4.6.1) manuell den 5 Gruppen „Gut“, „Schlecht“, „Sehr Schlecht“, „Sonstiges“ und „Kein Kopf“ zugeordnet. Die Frames wurden anhand der zur Normalisierung (Kapitel 3.5.3) virtuell erzeugten Grauwertbilder der oberen Industriekamera (Industriekamera 1 in Abbildung 3.6 aus Kapitel 3.3) markiert, in denen die Pferdeköpfe mit direktem Blick zur Kamera ausgerichtet sein sollten. Die Abbildungen 3.66 und 3.67 zeigen Beispieldaten der einzelnen Gruppen. Abbildung 3.66 zeigt Bilddaten, deren Dateien dem Tier „Rosi“ zugeordnet sind. Die Daten aus Abbildungen 3.67 sind dem Tier „Blüte“ zugeordnet.

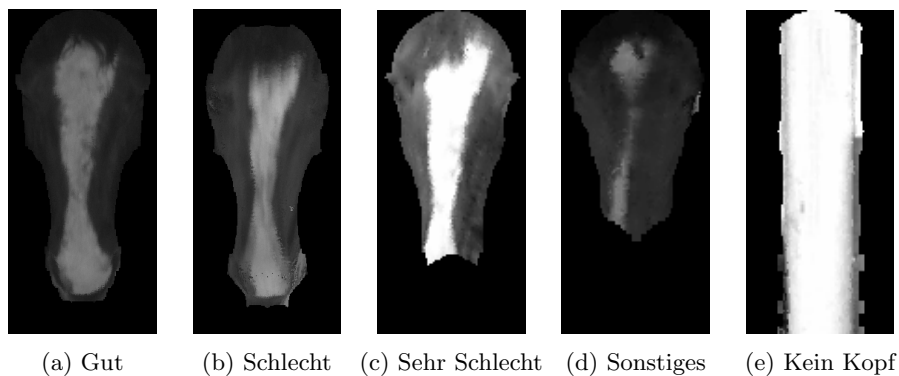


Abbildung 3.66.: Beispielgruppen (Rosi) der manuellen Gruppierung der Daten

Die Gruppe „Gut“ enthält solche Bilddaten, in denen eine Identifikation optimal ist: das Tier guckt direkt in die Kamera, es ist kein Hintergrund auf dem detektierten Bereich zu erkennen, der Pferdekopf ist nicht verdreht und vollständig. Die Gruppe „Schlecht“ enthält Bilddaten, die vermutlich zur Identifikation ausreichen würden, jedoch sind Teile vom Hintergrund sichtbar, Teile des Pferdekopfes fehlen (z.B. Nüstern oder Schopf in Abbildung 3.67b) oder der Kopf ist leicht verdreht (z.B. Abbildung 3.66b ist durch Teilverdeckung mit Hintergrund verdreht). Die Gruppe „Sehr Schlecht“ soll solche Bilddaten enthalten, in denen größere Teile des Pferdekopfes fehlen (siehe Abbildung 3.66c), die Pose eindeutig falsch erkannt wurde (siehe Abbildung 3.67c oder andere Teile des Tieres im Bild sichtbar sind, so dass eine Identifikation anhand der Pferdeköpfe fehlschlagen würde.

In der Gruppe „Sonstiges“ sind solche Aufnahmen geführt, in denen das Tier nicht dem durch die Datei zugeordneten Tier entspricht (Abbildung 3.66d zeigt „Delia“ anstatt „Rosi“ und 3.67d zeigt „Blütenzauber“ anstatt „Blüte“), das Tier einen Halfter trägt oder sonstige extreme Veränderungen im Bild zu erkennen sind. Die Gruppe „Kein Kopf“ fasst solche Bilddaten zusammen in denen kein Teil der Pferde zu

erkennen ist, sondern Teile aus dem Hintergrund als Pferdekopf detektiert wurden.

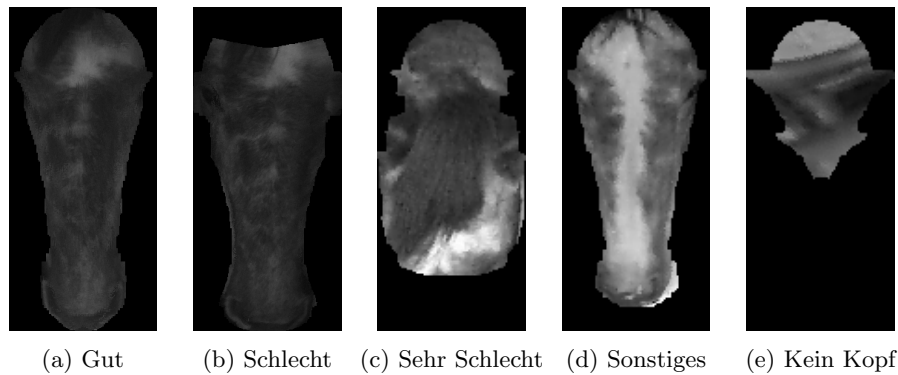


Abbildung 3.67.: Beispielgruppen (Blüte) der manuellen Gruppierung der Daten

Die Anzahl der aufgenommenen Frames jeder Gruppe ist im Kapitel 4.6.1 in den Tabellen 4.23 und 4.24 aufgelistet. Die Detektion aus Kapitel 3.4.4 wurde, wie bereits erwähnt, mit einer sehr schwachen Klassifikation mit einem Schwellwert von $T_{ROC} = 0,35$ für den Ähnlichkeitsfaktor ρ durchgeführt.

Daten zur Identifikation

Tabellen 4.23 und 4.24 aus Kapitel 4.6.1 führen die Anzahl der Frames auf, die in der Gruppe „Gut“ pro Tier vorhanden sind. Es wurde entschieden, dass mindestens 200 Frames dieser Gruppe zum Anlernen und Validieren der Identifikation pro Tier vorhanden sein sollten. Zudem wurde auf eine gleichmässige Beleuchtung geachtet, so dass lediglich solche Frames verwendet werden, bei denen beide Leuchtstoffröhren während der Aufnahmen in Betrieb waren. Die entsprechenden Frames, in denen die Pferde von beiden Seiten beleuchtet wurden, sind in der jeweiligen Zeile der Tabellen 4.23/4.24 in Grau hinterlegt.

Die Identifikation wurde mit einer Gruppe von 9 Tieren getestet. Für einige der Tiere existiert nur eine geringere Anzahl (keine 200 Frames) an Daten mit den festgelegten Kriterien, daher bleiben 5 der insgesamt 14 Pferde zur Validierung der Identifikation in Kapitel 4.6 unberücksichtigt.

Insgesamt wurden 1.800 Frames, pro Pferd 200 Frames, zufällig aus den grau hinterlegten Daten der Tabellen 4.23 und 4.24, welche der Gruppe „Gut“ (grüne Schrift in den Tabellen 4.23/4.24) zugeordnet sind, für die Auswertung in Kapitel 4.6.2 verwendet. Die Grauwerte der Bilddaten wurden mit der in Kapitel 3.6.1 beschriebenen Methode normalisiert. Nach der Normalisierung der Grauwerte wurden die Grauwertbilder maskiert, so dass Grauwertpixel, die in weniger als 300 der insgesamt 1.800 Frames verwendet werden, ausgeblendet werden.

Kreuzvalidierung

Zur Auswertung jeder Kamera wurden die 1.800 Frames in unterschiedlich große Teilmengen aufgeteilt. Die Größe der Teilmengen, welche im Anschluss durch eine Kreuzvalidierung aus den gesamten Daten als Testdatensatz ausgeschlossen wird, wurde verändert, um unterschiedlich große Lerndatensätze zu erzeugen. Ist K die Größe der jeweiligen Teilmenge, so wurde für jedes Tier die ersten K Datensätze (insgesamt $9K$ Datensätze) isoliert, um mit den $1.800 - 9K$ verbleibenden Datensätzen die Methode der Eigengesichter aus Kapitel 3.6.2 auf den normalisierten 2D Bilddaten anzulernen. Dies ist in der ersten Zeile der Abbildung 3.68 (Iteration #1) schematisch dargestellt.

	Tier #1					Tier #2					Tier #3				
Iteration #1	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Iteration #2	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Iteration #3	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Iteration #4	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■
Iteration #5	■	■	■	■	■	■	■	■	■	■	■	■	■	■	■

Lerndatensatz

Testdatensatz

Abbildung 3.68.: Schema Datenauswahl zur Kreuzvalidierung

Die roten Bereiche der Abbildung 3.68 stellen die gewählten Datensätze dar, welche zum Anlernen der Methode aus Kapitel 3.6.2 genutzt wurden. Anschließend wird mit den in Grün dargestellten Testdatensätzen der Klassifikator aus Kapitel 3.6.2 validiert. Die Abbildung 3.68 zeigt drei Tiere mit jeweils 5 Frames, von denen je $K = 2$ Frames zum Test der Methode verwendet wird. Der Anteil der Lerndatensätze in Abbildung 3.68 entspricht damit 60 % (vgl. Abbildungen 4.16 bzw. 4.17 und 4.18 aus Kapitel 4.6.2). Die Auswahl der Testdatensätze wird entsprechend der einzelnen Iterationen aus Abbildung 3.68 wiederholt, indem pro Tier ein Frame aus dem Trainingsdatensatz als Lerndatensatz verwendet wird und ein neuer Frame (zuvor im Lerndatensatz) in den Trainingsdatensatz aufgenommen wird. Pro Lerndatensatz ist die Anzahl der Daten pro Tier gleich und verändert sich nicht, wodurch die Auswertung einer k -fach stratifizierten Kreuzvalidierung entspricht (siehe *stratified cross-validation* in Kohavi [1995]). Die Kreuzvalidierung wird zudem auf mehrere Gruppierungen (nur ein Frame Versatz von Iteration # n zu # $n+1$ in Abbildung 3.68) angewendet, wodurch eine bessere Monte-Carlo Schätzung erreicht wird („*Repeating cross-validation multiple times using different splits into folds provides a better Monte-Carlo estimate to the complete cross-validation at an added cost.*“ nach Kohavi [1995]).

Auswertung der Daten

Die Klassifikation in Kapitel 3.6.2 wird anhand der Distanz $d^2(\boldsymbol{\Omega}', \hat{\boldsymbol{\Omega}}'_i)$ aus Gleichung 3.105 des zu prüfenden normalisierten Grauwertbild $\boldsymbol{\Omega}'$ mit den normalisierten Grauwertbildern $\hat{\boldsymbol{\Omega}}'_i$ des Lerndatensatzes im Gesichtsraum durchgeführt. Die Klassifikation aus Kapitel 3.6.2 ordnet ein Grauwertbild einem zuvor angelernten Pferd zu, welches die geringste Distanz $d^2(\boldsymbol{\Omega}', \hat{\boldsymbol{\Omega}}'_i)$ im Gesichtsraum aufweist. Die in Kapitel 4.6.2 zur Auswertung der Identifikationsergebnisse (1 : N Abbildung) verwendeten Kurven sind in Kapitel 3.8.3 beschrieben. Aus Gründen der Übersicht und des Umfangs werden die Erkennungsraten des ersten Ranges (ROR - *rank one recognition rate*) (entspr. Treffergenauigkeit (ACC), siehe Kapitel 3.8.3) in den Abbildungen 4.16 bzw. 4.17 für die im nächsten Abschnitt (*Auswertung mit Variationen*) beschriebenen Variationen zusammengestellt. Für eine Auswahl der Variationen des Unterkapitels *Auswertung mit Variationen* werden die in Kapitel 3.8.3 beschriebenen CMC-Kurven erzeugt. Die CMC-Kurven geben, neben der eigentlichen Klassifikation (geringste Distanz $d^2(\boldsymbol{\Omega}', \hat{\boldsymbol{\Omega}}'_i)$ im Gesichtsraum) des ersten Ranges, die Ergebnisse in ihrer Rangfolge (Ränge > 1) entsprechend der Ähnlichkeiten der Individuen in der Datenbank zum untersuchten Bild $\boldsymbol{\Omega}'$ an.

Zur Auswertung (Ausnahme sind die Erkennungsraten des ersten Ranges) wurde die jeweils kleinste Distanz $d^2(\boldsymbol{\Omega}', \hat{\boldsymbol{\Omega}}'_i)$ für jedes angelernte Tier bestimmt $d^2(\boldsymbol{\Omega}', \hat{\boldsymbol{\Omega}}'_a) = \min(d^2(\boldsymbol{\Omega}', \hat{\boldsymbol{\Omega}}'_i))$, wobei $\hat{\boldsymbol{\Omega}}'_i$ dem jeweiligen Tier a zugeordnet ist. Es werden für jedes Grauwertbild $\boldsymbol{\Omega}'$ insgesamt 9 (Anzahl der Tiere im Lerndatensatz) Distanzen $d^2(\boldsymbol{\Omega}', \hat{\boldsymbol{\Omega}}'_a)$ ($a \in [1, \dots, 9]$) bestimmt, die jeweils angeben wie ähnlich das entsprechende Grauwertbild dem einzelnen Pferd im Gesichtsraum ist.

Die Ergebnisse in den ROC- bzw. PR-Kurven lassen sich der Verifikation (1 : 1 Abbildung) zuordnen, wie es bereits in Kapitel 3.8.3 beschrieben wurde. Dazu werden die einzelnen Distanzen $d^2(\boldsymbol{\Omega}', \hat{\boldsymbol{\Omega}}'_a)$ zum jeweiligen Pferd a der Datenbank den zwei Klassen **p** (positiv) oder **n** (negativ) zugeordnet. Handelt es sich bei dem Bild $\boldsymbol{\Omega}'$ um das Pferd a , so wird die Klasse **p** (positiv) zugeordnet, alle restlichen Distanzen $d^2(\boldsymbol{\Omega}', \hat{\boldsymbol{\Omega}}'_a)$ erhalten die Markierung **n** (negativ). In Kapitel 4.6.2 ergibt sich daraus für jedes Tier eine Kurve in den Abbildungen 4.19-4.21. Durch dieses Vorgehen ist die Auswertung deutlich vereinfacht, da die Entscheidung zwischen mehreren Klassen (jedes Pferd entspricht einer Klasse) in die binäre Klassifikation zwischen **p** und **n** überführt wird. Die Auswertung der Identifikation mit einer Klasse pro Pferd wäre grafisch durch eine Vielzahl von Abbildungen möglich, würde jedoch den Umfang der Dissertation immens ausweiten. Das beschriebene Vorgehen wird in Fawcett [2006] vorgeschlagen, wobei auf eine verschobene Verteilung der Daten durch die zusätzlichen Elemente der Klasse **n** hinzuweisen ist. In dem beschriebenen Fall entspricht das Verhältnis der Daten in der Klassen a zu der Klasse „nicht a “ für alle Klassen (a) immer dem Verhältnis der Daten der Klassen **p** (positiv) zu **n** (negativ).

Durch die zuvor beschriebene k -fache Kreuzvalidierung, welche in Abbildung 3.68 dar-

3. Material und Methoden

gestellt wurde, entsteht eine Vielzahl an Kurven. Die Daten aus den einzelnen Kurven wurden durch das sortierte Zusammenführen in eine einzige Kurve überführt, wie es durch Fawcett [2006] für die ROC-Kurven (siehe Kapitel 3.8.2) beschrieben ist. Eine Alternative aus [Fawcett 2006] ist das vertikale Zusammenfügen der Kurven. Die entsprechenden Kurven der Abbildungen 4.19-4.21 (Kapitel 4.6.2) sind grafisch jedoch kaum von diesen zu unterscheiden, weshalb das vereinfachte Verfahren (das sortierte Zusammenführen der Daten) angewendet wird.

Auswertung mit Variationen

Zur Auswertung der Identifikation aus Kapitel 3.6.2 in Kapitel 4.6.2 wurden mehrere Variationen durchlaufen:

- Bildgröße (Auflösung)
- Anzahl der Eigenwerte
- Anteil der Lerndatensätze.

Die Identifikation wurde auf unterschiedliche Größen der normalisierten Grauwertbilder angewendet. Die Bilder wurden dazu auf die Auflösungen 10×22 (12,5 %), 20×44 (25 %), 30×66 (37,5 %) und 40×88 (50 %) von der ursprünglich extrahierten Bildgröße mit einer Auflösung von 80×175 (100 %) herunterskaliert.

Es wurden zur Identifikation eine unterschiedliche Anzahl von Eigenwerten (siehe Kapitel 3.6.2) verwendet. Die Anzahl der Eigenwerte bestimmen zugleich die Anzahl der Eigenvektoren (Basisvektoren), welche vorgeben wie gut die Rekonstruktion eines Gesichts machbar ist. Zudem bestimmt die Anzahl der Eigenvektoren die Dimension des in Kapitel 3.6.2 beschriebenen Gesichtsraums. Es wurden zur Identifikation 9, 18, 36 und 54 Eigenwerte eingesetzt.

Zuletzt wurde der Anteil der Lerndatensätze variiert, indem die Anzahl K der zur Kreuzvalidierung verwendenden Testdatensätze, wie bereits in diesem Kapitel beschrieben, unterschiedlich groß gewählt wurde. Der prozentuale Anteil der Lerndatensätzen, welcher in den Abbildungen aus Kapitel 4.6.2 angegeben ist, berechnet sich aus K nach $\frac{200-K}{200} 100\% = 100 - \frac{K}{2}\%$.

3.8. Datenaufbereitung und Datenauswertung

3.8.1. Datenauswertung mittels Boxplots

Quartile bieten die Möglichkeit Aussagen über Messdaten zu machen, deren Verteilung nicht der Normalverteilung entspricht. Im Fall der Normalverteilung werden das arithmetische Mittel μ (Mittelwert) und die Standardabweichung σ der Messwerte betrachtet. Dies ist jedoch weniger sinnvoll, sobald vermehrt Ausreißer auftreten, wodurch Mittelwert und Standardabweichung nur schwer interpretierbar bleiben [Krummenauer u. a. 2007]. Daher werden in diesem Kapitel Quartile und deren Darstellung in Boxplots, welche in einer deutschen Übersetzung auch Schachteldiagramme genannt werden [Buchecker u. a. 2011], eingeführt. Entsprechenden Einsatz finden sie in vielen Bereichen, von der Medizin [Krummenauer u. a. 2007], dem ökologischen Landbau [Buchecker u. a. 2011] bis hin zur Bioanalyse [Kammer 2012]

Quartile sind ein Spezialfall der Quantile [Bronštejn u. a. 2012, S. 825], welche die auszuwertenden Messdaten im Fall der Quartile in vier Bereiche aufteilen. Die Unterteilung geschieht mittels dem Median, dieser stellt bei einer ungeraden Anzahl von Messdaten den Wert in der Mitte der zuvor sortierten Messdaten dar [Kammer 2012]. Haben die Messdaten eine gerade Anzahl, wird der Median durch den Mittelwert der beiden mittleren Messwerte der zuvor geordneten Messwerte bestimmt. Durch den Median, welcher auch als 2. Quartil Q_2 bezeichnet wird, werden die gesamten Messdaten in zwei Bereiche aufgeteilt. Die Unterteilung dieser zwei Bereiche erfolgt wiederum durch den Median der Unterbereiche in insgesamt vier Bereiche. Liegt der Median (2. Quartil Q_2) genau auf einem der Messwerte (ungerade Anzahl), so wird das 1. Quartil Q_1 durch den Median der Messwerte, welche unterhalb der Position des 2. Quartils Q_2 liegen, bestimmt. Das 3. Quartil Q_3 ist in diesem Fall der Median der Messwerte oberhalb der Position des 2. Quartils Q_2 . Ist die Anzahl der Messwerte hingegen gerade, so liegt der Median (2. Quartil Q_2) nicht auf einem der Messwerte. Wie schon beschrieben wird der Median in einem solchen Fall durch den Mittelwert der beiden mittleren der sortierten Messwerte bestimmt. Die Zweiteilung der Messwerte passiert nun genau zwischen diesen beiden mittleren Messwerten, so dass der obere zu dem ersten Bereich und der untere zu dem zweiten Bereich gehört. Das 1. bzw. 3. Quartil (Q_1 bzw. Q_3) wird entsprechend durch den Median des unteren bzw. oberen Bereichs bestimmt.

Der 1. Quartil Q_1 wird auch als 25%-Quartil ($Q_{25\%}$) beschrieben, da unter ihm 25% der gesamten Messwerte liegen. Unter dem 2. bzw. 3. Quartil (Q_2 bzw. Q_3) liegen hingegen 50% bzw. 75% der gesamten Messwerte, entsprechend die Bezeichnung 50%- bzw. 75%-Quartil ($Q_{50\%}$ bzw. $Q_{75\%}$). Ausreißer werden erst bestimmt, nachdem die Quartile durch das Bestimmen des Medians (2. Quartil Q_2) der gesamten Messdaten und der Mediane Q_1 und Q_3 der durch den 2. Quartil Q_2 entstandenen Unterbereiche berechnet sind. Als Ausreißer werden solche Messwerte angesehen, die außerhalb eines Bereiches liegen, der durch einen vierfachen Interquartilsabstand

3. Material und Methoden

(IQA) beschränkt wird. Der IQA wird durch die Differenz von $Q_3 - Q_1$ berechnet und stellt einen Bereich dar in dem sich 50% der Messwerte befinden. Als Ausreißer werden Werte erachtet, die oberhalb von $Q_3 + 1,5 (Q_3 - Q_1)$ bzw. unterhalb von $Q_1 - 1,5 (Q_3 - Q_1)$ liegen, also sich jeweils dem 1,5fachen des IQA oberhalb von Q_3 bzw. unterhalb von Q_1 befinden.

In Abbildung 3.69 ist die Darstellung eines Boxplots gezeigt, der die Quantile, sowie Maximal- und Minimalwerte in einer Grafik anzeigt. Das 1. und 3. Quartil spannen die „Box“ auf, in der, wie bereits beschrieben, 50% der Messwerte liegen. Die Höhe der „Box“ entspricht dem eingezeichneten Interquartilsabstand (IQA), welcher lediglich zur Anschauung in die Grafik integriert ist. Es stellt ein Maß für die Streuung der Werte dar. Das 2. Quartil, also der Median, ist als mittlerer Balken, zwischen dem 1. und 2. Quartil, in der „Box“ eingezeichnet und teilt diese.

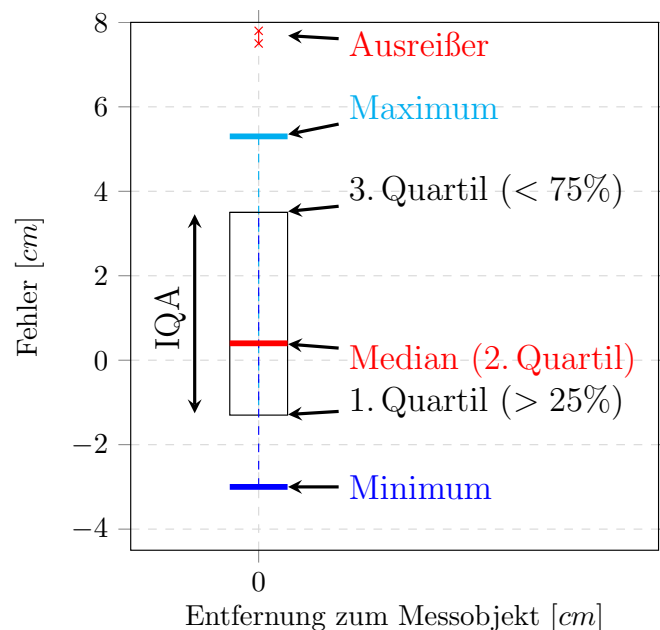


Abbildung 3.69.: Boxplot zur Auswertung der Messergebnisse

Der Median ist in der Abbildung 3.69 in rot dargestellt. Die Extrema (Minimum und Maximum) werden als Verlängerung aus dem Rechteck („Box“), welche sich durch Q_1 und Q_3 bildet, dargestellt und erinnern in der Form an Antennen. Im Englischen werden diese Fortsätze auch als „Whiskers“ bezeichnet. In Abbildung 3.69 ist das Maximum cyan und das Minimum in Blau dargestellt. Einige Ausreißer sind in roten Kreuzen über dem Maximum dargestellt, können aber auch unterhalb des Minimums auftreten. Das Maximum und das Minimum sind die entsprechenden Messwerte, welche nicht durch das zuvor beschriebene Vorgehen als Ausreißer detektiert wurden.

Je nach Art der Verteilung weicht der Median (Q_2) von dem Zentrum der „Box“ ab. Bei symmetrischer Verteilung liegt der Median genau in der Mitte und entspricht dem Mittelwert.

3.8.2. Datenauswertung mittels ROC-Grafik

ROC steht für „Receiver-Operator-Characteristics“, welches übersetzt so viel bedeutet wie „Sender-Empfänger-Charakteristik“ und auch als Grenzwertoptimierungskurve bezeichnet wird. Eine ROC-Grafik ermöglicht die Darstellung und den Vergleich zwischen Klassifikatoren anhand ihrer Performanz. So stellt dieses Kapitel eine Kurzfassung der Arbeiten durch Fawcett [2004, 2006]; Davis u. Goadrich [2006]; Irsoy u. a. [2012] und Majnik u. Bosnic [2013] dar. Ein Punkt in einer ROC-Grafik entspricht einem binären Klassifikator, welcher lediglich die diskrete Entscheidung zwischen zwei Klassen (hier **p** und **n**) zurückgibt. Die Vorhersage (Entscheidung) ist somit **p'** positiv oder **n'** negativ, wobei die tatsächliche Klasse des jeweiligen Elements unabhängig von der Vorhersage **p** positiv oder **n** negativ ist.

Grundlagen der ROC-Grafik

Die Basis zum Aufbau einer ROC-Grafik ist eine Wahrheitsmatrix (*engl: contingency table*; auch Konfusionsmatrix genannt - *engl: confusion matrix*). Für eine gegebene binäre Klassifikation (einzelner Punkt in ROC-Grafik) ergibt sich mit einem Satz von Testdaten eine Matrix, wie sie in Abbildung 3.70 dargestellt ist. In der Wahrheitsmatrix wird die Anzahl der jeweiligen Fälle eingetragen, welche sich durch die tatsächliche Klasse (Vergleichsdaten; Ground-Truth) und eine Vorhersage (Hypothese) durch den Klassifikator ergeben.

		Tatsächliche Klasse		
		p	n	
Vorhergesagte Klasse	p'	TP Richtig-Positiv	FP Falsch-Positiv	P'
	n'	FN Falsch-Negativ	TN Richtig-Negativ	N'
		P	N	

Abbildung 3.70.: Wahrheitsmatrix

Die Gesamtzahl der positiven Elemente ($\in \mathbf{p}$) im Satz der Testdaten ist P , die der

3. Material und Methoden

negativen Klasse ($\in \mathbf{n}$) N . Für einen (binären) Klassifikator ergibt sich die Anzahl der Elemente für die Vorhersage \mathbf{p}' (positiv) zu P' und entsprechend für \mathbf{n}' (negativ) zu N' . Die sich ergebenden vier Fälle sind TP (*engl: true positiv*) Richtig-Positiv, FN (*engl: false negative*) Falsch-Negativ, TN (*engl: true negative*) Richtig-Negativ und FP (*engl: false positiv*) Falsch-Positiv. Die diagonalen Einträge der Matrix stellen dabei eine richtige Entscheidung durch den Klassifikator dar, daher Richtig-Positiv (TP) oder Richtig-Negativ (TN). Die beiden übrigen Elemente sind entsprechende Fehlentscheidungen und sind Falsch-Positiv (FP) bzw. Falsch-Negativ (FN). Die Anzahl der jeweiligen Elemente in der Wahrheitsmatrix lassen sich spaltenweise zu

$$\begin{aligned} P &= TP + FN, \\ N &= FP + TN \end{aligned} \quad (3.123)$$

bzw. zeilenweise zu

$$\begin{aligned} P' &= TP + FP, \\ N' &= FN + TN \end{aligned} \quad (3.124)$$

aufsummieren.

Die Wahrheitsmatrix dient dazu unterschiedlichste Metriken für die jeweilige Klassifikation zu erstellen. Die Richtig-Positiv-Rate (TPR, *engl: true positive rate*) eines Klassifikators, welche auch Trefferquote (*engl: hit rate* bzw. *recall*), Sensitivität oder Empfindlichkeit (*engl: sensitivity*) genannt wird, berechnet sich nach Fawcett [2004, 2006]

$$TPR = \frac{TP}{P}. \quad (3.125)$$

Die Falsch-Positiv-Rate (FPR, *engl: false positive rate*), auch Ausfallrate (*engl: fallout*) oder Falschalarm-Rate (*engl: false alarm rate*) genannt, ist

$$FPR = \frac{FP}{N}. \quad (3.126)$$

Die Richtig-Negativ-Rate (TNR, *engl: true negative rate*), auch Spezifität (*engl: specificity*) genannt, berechnet sich zu

$$TNR = \frac{TN}{N} = 1 - FPR. \quad (3.127)$$

Der positive Vorhersagewert (PPV, *engl: positive predictive value*) wird auch Genauigkeit (*engl: precision*) genannt und berechnet sich durch

$$PPV = \frac{TP}{P'} = \frac{TP}{TP + FP}. \quad (3.128)$$

Die Treffergenauigkeit (ACC, engl.: *accuracy*) wird durch

$$ACC = \frac{TP + TN}{P + N} \quad (3.129)$$

bestimmt und stellt den Anteil der richtigen Entscheidungen (Diagonalelemente aus Abbildung 3.70) zur Gesamtzahl der Entscheidungen dar.

In der ROC-Grafik wird die Richtig-Positiv-Rate (TPR) einer Klassifikation auf der Ordinate gegen die Falsch-Positiv-Rate (FPR) auf der Abszisse aufgetragen. Abbildung 3.71 zeigt eine Darstellung einer ROC-Grafik mit fünf diskreten Klassifikatoren. Die untere linke Ecke (0, 0) entspricht dabei einem Klassifikator, der die Klasse \mathbf{p} ignorieren und durchgehend eine Vorhersage \mathbf{n}' zur Klasse \mathbf{n} trifft. Dagegen entspricht die obere rechte Ecke (1, 1) einem Klassifikator, welcher die Klasse \mathbf{n} ignoriert und durchgehend \mathbf{p}' vorhersagt. Die obere linke Ecke (0, 1) hingegen, in der sich der Klassifikator A befindet, entspricht einem perfekten Klassifikator. Denn je weiter nordwestlich sich ein Klassifikator in der ROC-Grafik befindet, um so besser ist er im Vergleich zu den übrigen Klassifikatoren.

Als „konservativ“ lassen sich solche Klassifikatoren bezeichnen, die weiter links, aber dafür näher an der X-Achse, in der Grafik angesiedelt sind. Je „konservativer“ ein Klassifikator, um so strenger wird die Klassifikation zur Klasse \mathbf{p} , wodurch die FPR fällt, jedoch häufig auch die TPR. Solche Klassifikatoren, welche weiter rechts ansiedeln, können dann als „liberal“ bezeichnet werden. Somit ist der Klassifikator C aus Abbildung 3.71 „konservativer“ als D.

Die diagonale Linie (Identitätslinie FPR=TPF), welche schraffiert dargestellt ist, stellt dabei solche Positionen dar, die Klassifikationen durch eine zufällige Schätzung erreichen. B ist ein entsprechendes Beispiel in Abbildung 3.71. Damit ein Klassifikator in die oberen linke Hälfte der Grafik und weg von der Diagonalen gelagt, muss er die Informationen aus den Daten zur Schätzung ausnutzen.

Solche Klassifikatoren unterhalb dieser Identitätslinie, wie E aus Abbildung 3.71, schätzen schlechter als zufällig. Wird deren Vorhersage jedoch umgekehrt, also $\mathbf{p}' \Rightarrow \mathbf{n}'$ und $\mathbf{n}' \Rightarrow \mathbf{p}'$, so wird der Klassifikator in die obere Hälfte transferiert. E aus Abbildung 3.71 ist schlechter als B und entspricht negiert dem Klassifikator D. E extrahiert damit nützliche Informationen aus den Daten, interpretiert diese jedoch falsch [Fawcett 2004, 2006].

ROC-Kurve

Bisher wurden diskrete Klassifikatoren beschrieben, die einem Punkt in der ROC-Grafik entsprechen. Ein Klassifikator, der eine kontinuierliche Zahl generiert, beschreibt mit dieser, wie ähnlich eine Instanz eines Datensatzes einer Klasse ist. Diese Zahl kann einer Wahrscheinlichkeit entsprechen mit der die Instanz einer Klasse angehört. Ansonsten entspricht die Zahl einer Rangfolge zwischen den Instanzen und

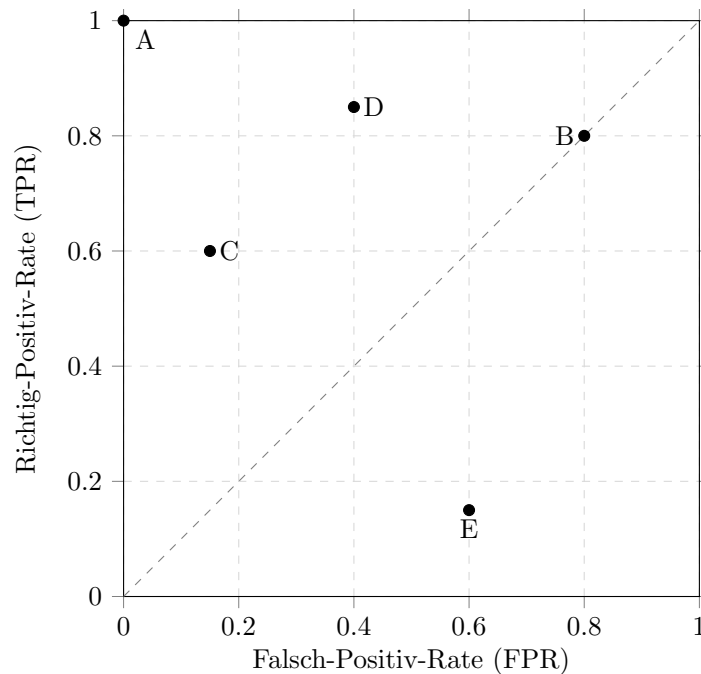


Abbildung 3.71.: ROC-Grafik mit fünf diskreten Klassifikatoren

kann als unkalibriert bezeichnet werden. Ein direkter Vergleich zwischen den Werten, welche durch unterschiedliche Klassifikatoren erzeugt werden, ist somit nur im kalibrierten Fall sinnvoll. Auf die ROC-Grafik hat eine solche Kalibrierung jedoch keinen Einfluss.

Durch den Einsatz eines Schwellwerts lässt sich ein solcher Klassifikator, der eine kontinuierliche Zahl erzeugt, in einen diskreten Klassifikator überführen. Ist der Wert des kontinuierlichen Klassifikators kleiner gleich dem Schwellwert, entspricht dies der binären Schätzung \mathbf{p}' , ansonsten \mathbf{n}' . Unterschiedliche Schwellwerte erzeugen so unterschiedliche Punkte auf der ROC-Grafik. Die einzelnen Punkte ergeben sich, durch das Durchlaufen mittels dem Schwellwert, zu einer Kurve, wie sie in Abbildung 3.72 dargestellt ist.

Tabelle 3.11 zeigt einen Datensatz, mit dem das Erstellen der ROC-Grafik aus Abbildung 3.72 erklärt werden soll. Das Beispiel ist bereits in aufsteigender Reihenfolge nach den Werten des Klassifikators sortiert, ist dies nicht der Fall wird der Datensatz zunächst entsprechend sortiert.

Der binäre Klassifikator wurde entgegen der Vorlage aus Fawcett [2004] bzw. Fawcett [2006] definiert. Der Klassifikator trifft dort die Vorhersage \mathbf{p}' , wenn der Wert der Instanz eines Datensatzes größer dem Schwellwert ist. Und im umgekehrten Fall \mathbf{n}' .

Tabelle 3.11.: Beispieldatensatz zur Beschreibung der ROC-Grafik

Element	Klasse	Wert	Element	Klasse	Wert
1	p	0,1	6	n	0,5
2	p	0,2	7	n	0,6
3	n	0,3	8	p	0,7
4	p	0,4	9	n	0,8
5	p	0,5	10	n	0,9

Die Werte des Datensatzes sind entsprechend in abfallender Reihenfolge sortiert. Für jeden einzelnen Punkt bedeutet dies ein Vertauschen der Zeilen der Wahrheitsmatrix. Der durch Fawcett [2004, 2006] beschriebene Klassifikator gibt einen Wert zurück, welcher die Ähnlichkeit zur Klasse **p** beschreibt, dieser wird höher, je wahrscheinlicher der Klassifikator die Instanz für ein Element der Klasse **p** hält. Dagegen beschreibt der zuvor erwähnte Klassifikator eine Differenz, die höher wird je unwahrscheinlicher die Instanz der Klasse **p** angehört. Da die Werte ohnehin unkalibriert sind, lässt sich ein der Differenz entsprechender Wert durch seinen Kehrwert in einen der Ähnlichkeit entsprechenden Wert überführen.

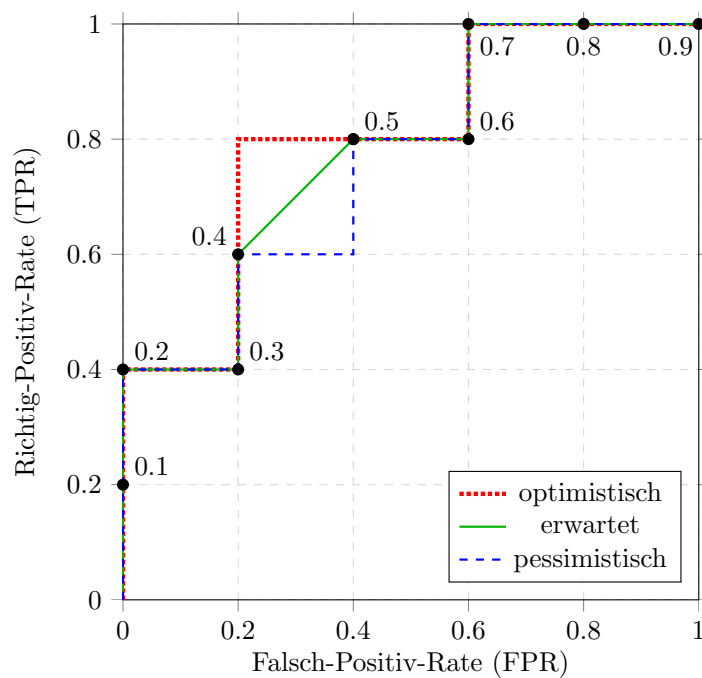


Abbildung 3.72.: ROC-Grafik mit Kurven aus kontinuierlichen Zahlen eines Klassifikators

3. Material und Methoden

Da die ROC-Grafik lediglich die relativen Werte betrachtet, hat eine Änderung der Werte keinen Einfluss auf die Grafik, solange die Reihenfolge der Instanzen nach dem Sortieren unverändert bleibt. In Tabelle 3.11 sind jeweils 5 Elemente der Klasse **p** ($P = 5$) und 5 Elemente der Klasse **n** aufgeführt ($N = 5$). Die Anzahl P (Anzahl der Elemente in der Klasse **p**) gibt die Schrittweite $\frac{1}{P}$ der Punkte auf der Ordinate an. Die entsprechende Schrittweite auf der Abszisse beträgt $\frac{1}{N}$. Beim Durchlaufen des nach den Werten des Klassifikators sortierten Datensatzes, wird die TPR um die Schrittweite $\frac{1}{P}$ erhöht, wenn es sich um ein Element der Klasse **p** handelt. Im Fall, dass die aktuell zu prüfende Instanz Element der Klasse **n** ist, wird die FPR um die Schrittweite $\frac{1}{N}$ erhöht. Gestartet wird in der linken unteren Ecke, also mit $TPR = 0$ und $FPR = 0$. In Abbildung 3.72 sind die einzelnen Punkte durch schwarze Kreise gekennzeichnet. Zudem sind neben den Punkten die zugehörigen Schwellwerte eingezeichnet. Einen Sonderfall stellen Instanzen mit einem gleichen Wert dar. In dem Datensatz aus Tabelle 3.11 stimmen die Werte der Instanzen 5 und 6 überein. Dadurch, dass es sich um ein Element der Klasse **p** und ein Element der Klasse **n** handelt, ist ein diagonaler Schritt beim Schwellwert 0,5 in der Abbildung 3.72 zu erkennen. Zusammenhängend mit der Reihenfolge, in der die jeweiligen Klassen abgearbeitet werden, ergeben sich mehrere Fälle. Der optimistische Fall ist in rot mit einer gepunkteten Linie eingezeichnet, es werden hierzu erst die Schritte der Klasse **p** berücksichtigt. Mit der gestrichelten blauen Linie ist der pessimistische Fall gekennzeichnet, welcher zuerst die Schritte der Klasse **n** durchführt. Mit der grünen Linie ist der Durchschnitt (erwartete Kurve) der optimistischen und pessimistischen Werte markiert, dieser entspricht einem gleichzeitigen Schritt ohne eine Klasse zu bevorzugen. In der vorliegenden Arbeit werden zur Darstellung die erwartete Kurve aus dem Durchschnitt der optimistischen und der pessimistischen Kurve dargestellt.

Fläche unter ROC-Kurve

Die Fläche (AUC - *engl: area under curve*) unter der ROC-Kurve dient dazu einen Klassifikator, dabei ist egal ob er diskrete Entscheidungen oder eine kontinuierliche Zahl zurückgibt, mit nur einem Wert in seiner Leistungsfähigkeit beschreiben zu können. Im Unterkapitel *Grundlagen der ROC-Grafik* wurde bereits dargelegt, dass je weiter eine ROC-Kurve zur oberen linken Ecke $(0, 1)$ tendiert, umso besser ist der Klassifikator einzuschätzen. Dies steht im direkten Zusammenhang mit der Fläche, welche sich unterhalb der ROC-Kurve befindet.

Für die Werte aus Tabelle 3.11 ergeben sich die Punkte aus Abbildung 3.72, deren Koordinaten nochmals in Tabelle 3.12 aufgeführt sind.

Tabelle 3.12.: Punktkoordinaten in der ROC-Grafik

Element	1	2	3	4	5/6	7	8	9	10
FPR	0,0	0,0	0,2	0,2	0,4	0,6	0,6	0,8	1,0
TPR	0,2	0,4	0,4	0,6	0,8	0,8	1,0	1,0	1,0

Die Fläche unter der Kurve kann nun durch

$$AUC = \sum_{n=1}^{P+N-1} (FPR(n+1) - FPR(n)) \frac{TPR(n+1) + TPR(n)}{2} \quad (3.130)$$

berechnet werden. Die gesamte Fläche unter der Kurve teilt sich in einzelne Trapeze, welche über die gesamte Fläche aufaddiert werden. Die Differenz $FPR(n+1) - FPR(n)$ ist dabei der Schritt entlang der Abszisse, welche der Grundlinie des jeweiligen Trapezes entspricht. Mit $\frac{TPR(n+1)+TPR(n)}{2}$ wird dann die Länge der Mittellinie des Trapezes bestimmt [Bronštejn u. a. 2012, S. 139]. In den Fällen $TPR(n+1) = TPR(n)$ handelt es sich bei den Trapezen um Rechtecke, wodurch der Durchschnitt $\frac{TPR(n+1)+TPR(n)}{2} = TPR(n+1) = TPR(n)$ gleichzeitig der Höhe des Rechtecks entspricht. Lediglich in den Fällen, in denen die nebeneinander liegenden Punkte unterschiedliche Höhen haben, wie dies vom Element 4 zu 5 in Abbildung 3.72 erkennbar ist, wird tatsächlich ein Trapez zur Berechnung der Fläche benötigt. Die Fläche des Trapezes ergibt sich durch die Multiplikation der Grundlinie mit der Mittellinie [Bronštejn u. a. 2012, S. 139]. Gleiche Punkte, wie die Punkte von Element 5 und 6, sowie übereinanderliegende Punkte, wie die der Elemente 1 und 2, bleiben durch die Schrittweite Null in der Abszisse bei der Flächenberechnung unberücksichtigt. Die Fläche unter der ROC-Kurve des Datensatzes aus Tabelle 3.11 ergibt sich damit zu $AUC = 0,78$.

Die Werte der AUC liegen immer zwischen 0 und 1, wobei Werte unterhalb von 0,5 (0,5 entspricht der AUC der Identitätslinie) für eine schlechte Klassifikation sprechen. Die AUC gibt dabei ein Maß für die durchschnittliche Leistungsfähigkeit eines Klassifikators an, wobei ein Klassifikator mit einer größeren AUC einem zweiten in bestimmten Bereichen der ROC-Kurve unterlegen sein kann. Die Besonderheit der AUC ist ihre Unabhängigkeit von der angenommenen Verteilung der Daten [Majnik u. Bosnic 2013]. Weitere Variationen der AUC werden durch Majnik u. Bosnic [2013] beschrieben, werden in der vorliegenden Dissertation jedoch nicht weiter berücksichtigt.

PR-Kurve

In der PR-Kurve (*precision-recall curve*) wird die Genauigkeit PPV (*engl. precision*, siehe Gleichung 3.128) gegen die schon in der ROC-Kurve verwendete Trefferquote TPR (Sensitivität oder Empfindlichkeit - *engl. recall*, siehe Gleichung 3.125) aufgetragen. Der wesentliche Unterschied der PR-Kurve zur ROC-Kurve ist, dass das Verhältnis zwischen den Richtig-Positiven TP und Falsch-Positiven FP Fällen (siehe Abbildung 3.70) durch die Genauigkeit (PPV) besser abgebildet wird. Vorrangig bei Datensätzen, deren Verteilung zwischen den Klassen **p** und **n** schief ist, so dass die Anzahl N deutlich größer ist als die Anzahl P , ist dies von Vorteil. Unterschiede zwischen Klassifikatoren lassen sich teilweise nicht in der ROC-Kurve ausmachen, sind jedoch in der PR-Kurve deutlich erkennbar. Nach Davis u. Goadrich [2006]

3. Material und Methoden

dominiert ein Klassifikator nur in der ROC-Kurve, wenn er auch in der PR-Kurve dominiert. Dabei ist ein Klassifikator in der PR-Kurve besser als ein anderer je näher die Kurve dem rechten oberen Punkt kommt und nicht, wie bei der ROC-Kurve, dem linken oberen. Dadurch wirkt die PR-Kurve im Vergleich zur ROC-Kurve horizontal gespiegelt. Für jeden Punkt der ROC-Kurve lässt sich ein entsprechender Punkt in der PR-Kurve bestimmen, die Punkte dazwischen werden nicht linear nach Davis u. Goadrich [2006] interpoliert.

3.8.3. CMC-Kurve

Kapitel 3.8.2 beschreibt die Auswertung einer zwei-Klassen (binären) Entscheidung. Dies entspricht in der Gesichtserkennung einer Verifikation, welche ein Gesicht auf eine Identität prüft, also ein 1:1 Abgleich ist. Das Gesicht kann entweder zum überprüften Individuum gehören (z.B. Klasse \mathbf{p}) oder eben nicht (z.B. Klasse \mathbf{n}). Die Identifikation durch die Gesichtserkennung ist jedoch eine mehr-Klassen Entscheidung (hier zwischen den Klassen \mathbf{c}_k) und gleicht ein Bild mit den Individuen in der Datenbank ab, um eine Zuordnung treffen zu können. Die Identifikation durch Gesichtserkennung ist damit ein 1: K Ableich (z.B. *Identifikation der einzelnen Individuen bei Pferden* siehe Kapitel 3.6), da eine Zuordnung zwischen mehreren Klassen (nicht binär) entschieden wird.

Die Konfusionsmatrix (Wahrheitsmatrix) aus Abbildung 3.70 wird für eine Klassifikation zwischen N_C Klassen \mathbf{c}_k zu einer $N_C \times N_C$ Matrix, wie sie in Abbildung 3.73 dargestellt ist. In jeder Spalte sind die tatsächlichen Klassen \mathbf{c}_k ($k \in [1, \dots, K]$) aufgeführt, in den Zeilen sind die jeweiligen Vorhersagen der Klassifikation \mathbf{c}'_k zu finden.

		Tatsächliche Klasse					
		\mathbf{c}_1	\mathbf{c}_2	\mathbf{c}_3	\dots	\mathbf{c}_k	
Vorhergesagte Klasse	\mathbf{c}'_1	\mathbf{TP}_1	\mathbf{E}_{21}	\mathbf{E}_{31}	\dots	\mathbf{E}_{k1}	N'_1
	\mathbf{c}'_2	\mathbf{E}_{12}	\mathbf{TP}_2	\mathbf{E}_{32}	\dots	\mathbf{E}_{k2}	N'_2
	\mathbf{c}'_3	\mathbf{E}_{13}	\mathbf{E}_{23}	\mathbf{TP}_3	\dots	\mathbf{E}_{k3}	N'_3
	\vdots			\vdots			
	\mathbf{c}'_k	\mathbf{E}_{1k}	\mathbf{E}_{2k}	\mathbf{E}_{3k}	\dots	\mathbf{TP}_k	N'_k
		N_1	N_2	N_3		N_k	

Abbildung 3.73.: Konfusionsmatrix einer mehr-Klassen Entscheidung

So ist N_k die Anzahl der Testdaten, die zu der jeweiligen Klasse \mathbf{c}_k gehören. Entsprechend ist N'_k die Anzahl der Testdaten für die eine Klassenzugehörigkeit zur Klasse \mathbf{c}_k vorhergesagt wird \mathbf{c}'_k . Die Diagonalelemente TP_k der Konfusionsmatrix aus Abbildung 3.70 entsprechen der Anzahl von Richtig-Positiv Entscheidungen für die jeweilige Klasse \mathbf{c}_k . Die Einträge E_{ij} ($i, j \in [1, \dots, K]$), welche sich nicht auf der Diagonalen der Konfusionsmatrix befinden, sind Fehlentscheidungen.

Die Entscheidung der mehr-Klassen Klassifikation lässt sich für jede Klasse \mathbf{c}_k als eine binäre Klassifikation (nach Kapitel 3.8.2) beschreiben. Die Anzahl der positiven Entscheidungen \hat{P}_k (vgl. P aus Kapitel 3.8.2) entspricht dann N_k , die der negativen Entscheidungen pro Klasse

$$\hat{N}_k = \sum_{i \neq k} N_i \quad . \quad (3.131)$$

Entsprechend lassen sich \hat{P}'_k und \hat{N}'_k (entspricht P' bzw. N' aus Kapitel 3.8.2) für jede Klasse nach

$$\hat{P}'_k = N'_k \quad \text{bzw.} \quad \hat{N}'_k = \sum_{i \neq k} N_i \quad (3.132)$$

bestimmen. Für jede Klasse \mathbf{c}_k wird die Anzahl der Richtig-Negativ Entscheidungen (TN_k) durch

$$TN_k = \sum_{i \neq k} TP_i + \sum_{i, j \neq k} E_{ij} \quad (3.133)$$

3. Material und Methoden

bestimmt. Die Anzahl der Falsch-Positiv Entscheidungen (FP_k) ergibt sich aus

$$FP_k = \sum_{j=k} E_{ij} \quad (3.134)$$

und die Anzahl der Falsch-Negativ Entscheidungen (FN_k) durch

$$FN_k = \sum_{i=k} E_{ij} \quad (3.135)$$

Um die Trefferquote (TPR), die Ausfallrate (FPR), die Spezifität (TNR) oder die Genauigkeit (PPV) je Klasse \mathbf{c}_k zu berechnen, werden die Gleichungen 3.125-3.128 aus Kapitel 3.8.2 mit den beschriebenen Entscheidungswerten (TP_k , TN_k , FP_k , FN_k) genutzt. Diese lassen sich, wie zuvor in Kapitel 3.8.2 beschrieben, als Punkte einer diskreten Klassifikation (Unterkapitel *ROC-Kurve* des Kapitels 3.8.2) in die ROC- bzw. PR-Kurven eintragen. Die bisherigen ROC- und PR-Kurven aus Kapitel 3.8.2 lassen sich einzeln für jede Klasse \mathbf{c}_k erstellen. Dazu wird ein kontinuierlicher Wert benötigt, der die Ähnlichkeit des einzutragenden Testsatzes zur jeweiligen Klasse \mathbf{c}_k beschreibt. Hierzu lässt sich zum Beispiel die Distanz $d^2(\boldsymbol{\Omega}', \hat{\boldsymbol{\Omega}}'_i)$ (nach Gleichung 3.105) aus Kapitel 3.6.2 zu der jeweiligen Klasse \mathbf{c}_k verwenden, wobei $\boldsymbol{\Omega}'$ dem zu prüfenden Datensatz entspricht und nur die Bilddaten $\hat{\boldsymbol{\Omega}}'_i$ der Klasse \mathbf{c}_k berücksichtigt werden. Um die mehr-Klassen Entscheidung in einer einzigen Kurve darzustellen, reichen die Kurven aus Kapitel 3.8.2 (*ROC-Kurve* bzw. *PR-Kurve*) nicht aus.

Die Treffergenauigkeit (ACC, *engl: accuracy*) wird entgegen der Gleichung 3.129 für die mehr-Klassen Entscheidung durch

$$ACC = \frac{\sum_{i=1}^K TP_i}{\sum_{i=1}^K N_i} \quad (3.136)$$

bestimmt und stellt, wie zuvor in Kapitel 3.8.2, den Anteil der richtigen Entscheidungen (Diagonalelemente aus Abbildung 3.73) zur Gesamtzahl der Entscheidungen dar.

Die durch Gleichung 3.136 gegebene Treffergenauigkeit wird von Štruc u. Pavešić [2010] als die Erkennungsrate des ersten Ranges (ROR - *rank one recognition rate*) bezeichnet. Die CMC-Kurve (*cumulative match curve*), wie sie auch in [Phillips u. a. 2000] zur Beurteilung von Gesichtserkennungs-Algorithmen eingesetzt wird, stellt die Erkennungsraten gegen den jeweiligen Rang dar. Dabei wird eine Rangliste, ähnlich dem Ergebnis einer Suchmaschine, anhand der Ähnlichkeit (z.B. Distanz nach Gleichung 3.105 aus Kapitel 3.6.2) zu jedem Individuum der Datenbank zunächst in einer sortierten Liste eingetragen. Die größte Ähnlichkeit, also der erste Eintrag ($r = 1$, erster Rang) dieser Liste, entspricht dabei dem Klassifikationsergebnis \mathbf{c}'_k . Werden mehrere Ränge berücksichtigt, also neben dem besten Ergebnis ($r = 1$) beispielsweise auch das zweitbeste Ergebnis ($r = 2$), wird für eine Reihe von Testdaten geprüft,

wie häufig die tatsächliche Klasse c_k unter den ersten r Rängen in der sortierten Liste aufzufinden ist.

Abbildung 3.74 zeigt beispielhaft eine CMC-Kurve (*cumulative match curve*), in der die eingetragene Treffergenauigkeit stetig ansteigt, so dass diese nur gleich oder größer der Treffergenauigkeit der vorhergehenden Ränge sein kann. Die Ergebnisse sind besser, um so steiler die CMC-Kurve (*cumulative match curve*) ansteigt. Symbolisch ist dazu in Abbildung 3.74 ein roter Pfeil eingezeichnet, denn je steiler die Kurve steigt um so näher liegt die Kurve am oberen linken Punkt (ACC 100 % bei Rang 1), so dass die Ergebnisse, wie auch bei den ROC-Kurven (vgl. Abbildung 3.71) aus Kapitel 3.8.2, besser sind je näher die Kurven diesem Punkt kommen.

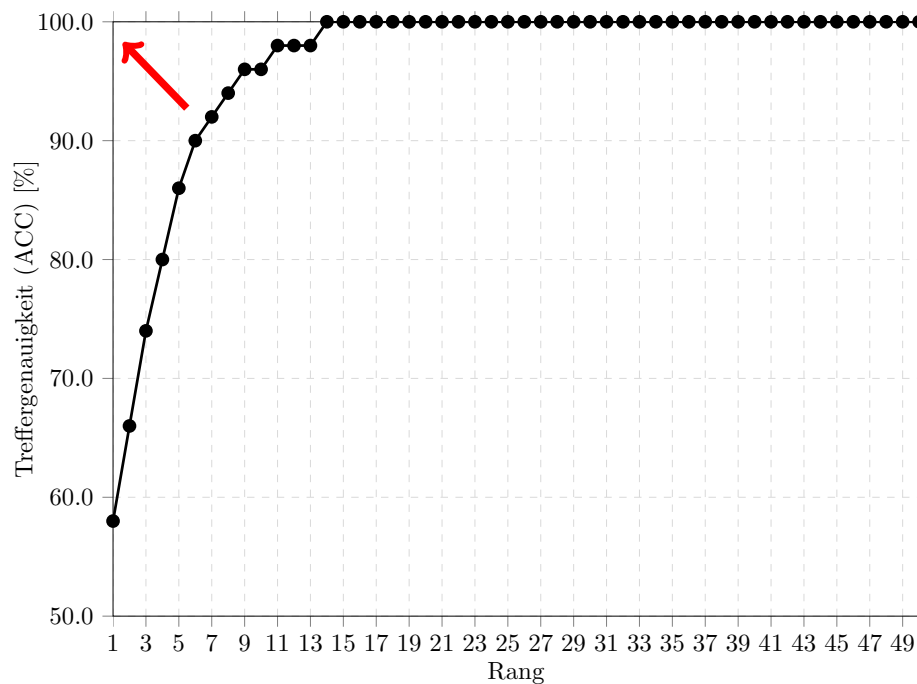


Abbildung 3.74.: Beispiel einer CMC-Kurve

In den Arbeiten von Bolle u. a. [2005] und DeCann u. Ross [2013] werden die Beziehungen zwischen den ROC- (Kapitel 3.8.2) und CMC-Kurven beschrieben. In DeCann u. Ross [2013] wird empfohlen die ROC-Kurven (Kapitel 3.8.2), trotzdem diese die Ergebnisse einer Verifikation beschreiben, zu den CMC-Kurven zur Beschreibung von Identifikationssystemen mit anzugeben.

4. Ergebnisse

4.1. Aufnahme von 2D/3D Daten in einer Versuchsstation

Die bereits im Kapitel 3.3 ausführlich beschriebene Aufnahmeeinheit wurde im Versuchsstall der Firma HIT (Hinrichs Innovation + Technik GmbH) aus Kapitel 3.2 dazu genutzt eine Datenbank an Messaufnahmen der Tiere aus Kapitel 3.1 zu erstellen. Die für die Messdaten dieser Dissertation entwickelte Aufnahmeeinheit wurde in Abbildung 3.6 aus Kapitel 3.3 gezeigt.

Erste Ergebnisse zur Aufnahmeeinheit wurden in Stahl u. a. [2012] veröffentlicht. Die Aufnahmeeinheit ermöglicht eine Aufnahme einer videoartigen Sequenz der Tiere mit bis zu 7 Frames pro Sekunde. Ein Frame besteht, wie bereits in Kapitel 3.7.2 beschrieben, aus mehreren synchron erfassten Bilddaten: dem Tiefenbild (640×480), einem Farbbild (640×480) und zwei Grauwertbildern (1600×1200).

Die Besuche der Tiere wurden entsprechend der Tabelle 3.4 aus Kapitel 3.7.2 mit 5 Frames pro Sekunde aufgezeichnet. Tabelle 4.1 fasst die Anzahl der aufgenommenen Frames zusammen. In jedem Feld ist zudem die Anzahl der Besuche aus Tabelle 3.4 in Klammern aufgeführt. In den Zeilen der Tabelle 4.1 sind die einzelnen Aufnahmezeiten (Spalten 1 „von“ und 2 „bis“) aufgelistet. In den Spalten 3 und 4 ist zudem die Art der Beleuchtung gekennzeichnet. Die restlichen Spalten der Tabelle 4.1 (ab Spalte 5) stehen für die jeweiligen Tiere aus Kapitel 3.1.

4. Ergebnisse

Tabelle 4.1.: Übersicht der Anzahl erfasster Frames (Besuche) pro Tier

Zeitraum		Beleuchtung		rechts	Anna	Arielle	Baldo	Bara	Blüte	Blütenzauber	Delia	Diadem	Dumblodore	Linus	Romanze	Rosi	Royal Bess	Serenade
von	bis		links															
10.11. 2011	10.11. 2011	x	x	x	-	3,4k (1)	-	802 (1)	-	-	-	-	-	-	-	-	-	-
13.04. 2012	13.04. 2012	x	x	x	278 (1)	620 (1)	-	655 (1)	573 (3)	378 (1)	-	136 (1)	-	-	615 (1)	504 (5)	-	-
24.07. 2012	11.08. 2012	x	x	x	-	-	1,7k (4)	-	-	-	-	-	-	-	-	10,6k (13)	9,4k (14)	10,8k (17)
10.12. 2012	02.01. 2013	x	x	x	12,9k (22)	13,9k (25)	-	8,0k (14)	14,8k (25)	11,9k (21)	13,6k (28)	-	7,7k (13)	-	14,9k (23)	13,3k (23)	-	-
04.01. 2013	10.01. 2013	x	x		12,3k (23)	-	-	9,8k (20)	7,8k (16)	8,8k (18)	9,1k (18)	-	10,1k (19)	-	9,1k (16)	6,0k (12)	-	-
11.01. 2013	27.01. 2013			x	7,3k (11)	11,1k (18)	-	10,1k (17)	10,9k (20)	8,1k (13)	11,6k (21)	-	8,3k (14)	-	8,4k (12)	10,1k (19)	-	-
28.01. 2013	28.02. 2013	x	x	x	23,4k (46)	21,9k (39)	-	19,2k (35)	20,1k (39)	17,1k (32)	32,3k (65)	-	21,6k (40)	-	20,4k (34)	19,9k (38)	-	-
23.04. 2013	26.04. 2013	x	x		-	2,8k (4)	-	2,6k (4)	-	1,9k (2)	1,7k (4)	-	871 (2)	3,2k (9)	1,8k (2)	2,5k (4)	-	-
26.04. 2013	30.04. 2013			x	-	1,9k (4)	-	1,6k (4)	-	1,5k (3)	1,4k (2)	-	2,1k (4)	1,9k (4)	1,4k (3)	655 (1)	-	-
30.04. 2013	06.05. 2013	x	x	x	-	9,8k (17)	-	-	-	-	7,2k (13)	-	6,3k (12)	2,3k (4)	5,7k (9)	6,6k (12)	-	-
06.05. 2013	07.05. 2013			x	-	2,6k (5)	-	-	-	-	2,3k (5)	-	3,5k (7)	2,3k (5)	1,2k (3)	2,5k (5)	-	-

4.2. Ergebnisse zur Messgenauigkeit der Tiefenkamera

Die Beschreibung des Aufbaus der hier beschriebenen Auswertungen findet sich im Unterbereich *Hardware Aufbau* von Kapitel 3.7.1.

Tiefenmessung 1

Der Ablauf zur Messung der hier beschriebenen Ergebnisse ist im Unterbereich *Tiefenmessung 1* von Kapitel 3.7.1 beschrieben. Die Auswertung in Abbildung 4.1 zeigt die Fehler zu der in jeder Position eingepassten Ebene mit der in Kapitel 3.8.1 vorgestellten Boxplots. Die obere Hälfte zeigt dabei den gesamten Fehlerbereich von $12,5\text{ cm}$ bis $-12,5\text{ cm}$, die untere Hälfte zeigt den Bereich von $2,5\text{ cm}$ bis $-2,5\text{ cm}$, auf dem die blau dargestellten Bereiche zwischen dem ersten Q_1 und dem dritten Quartil Q_3 besser zu erkennen sind. Der in Blau gekennzeichnete Bereich (zwischen Q_1 und Q_3) in beiden Darstellungen der Boxplots aus Abbildung 4.1 stellt dabei die Verteilung von 50 % der Daten (siehe Kapitel 3.8.1) in der entsprechenden Entfernung dar. In rot sind jeweils die Mediane (Q_2) dargestellt. Die schwarzen Antennen stellen die Werte dar, welche sich innerhalb des vierfachen Interquartalsabstands (IQA) ausgehend vom Median (Q_2) befinden. In der durchgeführten Messreihe liegen (mit geringen Ausnahmen, in denen der IQA im Verhältnis sehr klein wird) über 99% der Daten innerhalb der durch die schwarzen Antennen begrenzten Bereiche. Die in Magenta dargestellten Kreuze sind das absolute Minimum und Maximum der Daten, welche als Ausreisser außerhalb der schwarzen Antennen sichtbar sind.

Für jede der 200 Aufnahmen standen 22401 Disparitätenwerte, welche durch die Maske aus Abbildung 3.55 (siehe Kapitel 3.7.1) bestimmt sind, zur Verfügung. Die Auswertung besteht daher aus knapp 4,4 Millionen Messerwerten pro Entfernung (1 cm Schritte von $32,4\text{ cm}$ – $382,4\text{ cm}$). Entsprechend der Angaben aus dem Teilbereich *Tiefenmessung 1* des Kapitels 3.7.1 sind Messwerte unter einem Abstand von 70 cm nur vereinzelt verwertbar, so dass diese als nicht zuverlässig anzusehen sind. Die Entfernung der pro Position eingepassten Ebenen variiert zwischen $\pm 200\text{ }\mu\text{m}$ und wird daher nicht weiter berücksichtigt.

In Abbildung 4.1 ist zu erkennen, dass der Median in jeder Entfernung zentral bei Fehlerposition 0 cm liegt. Die Absolutwerte der Mediane im gesamten Messbereich sind kleiner $350\text{ }\mu\text{m}$.

Tabelle 4.2 fasst die Fehlerbereiche, welche innerhalb der in der ersten Spalte angegebenen Entfernungen auftreten, nochmals zusammen. Dazu ist in der zweiten Spalte der maximal auftretende Interquartalsabstand (IQA) angegeben, in dem sich 50% der Daten befinden (blauer Bereich aus Abbildung 4.1). Die dritte Spalte gibt die Spannweite des Fehlers, welcher durch die schwarzen Antennen in Abbildung 4.1 dargestellt ist, an. Innerhalb dieses in der Tabelle angegebenen Bereiches liegen (ohne die zuvor erwähnten Ausnahmen) 99 % der Daten. Die vierte Spalte gibt die Differenz zwischen maximalem und minimalem Ausreisser an, welcher innerhalb der

4. Ergebnisse

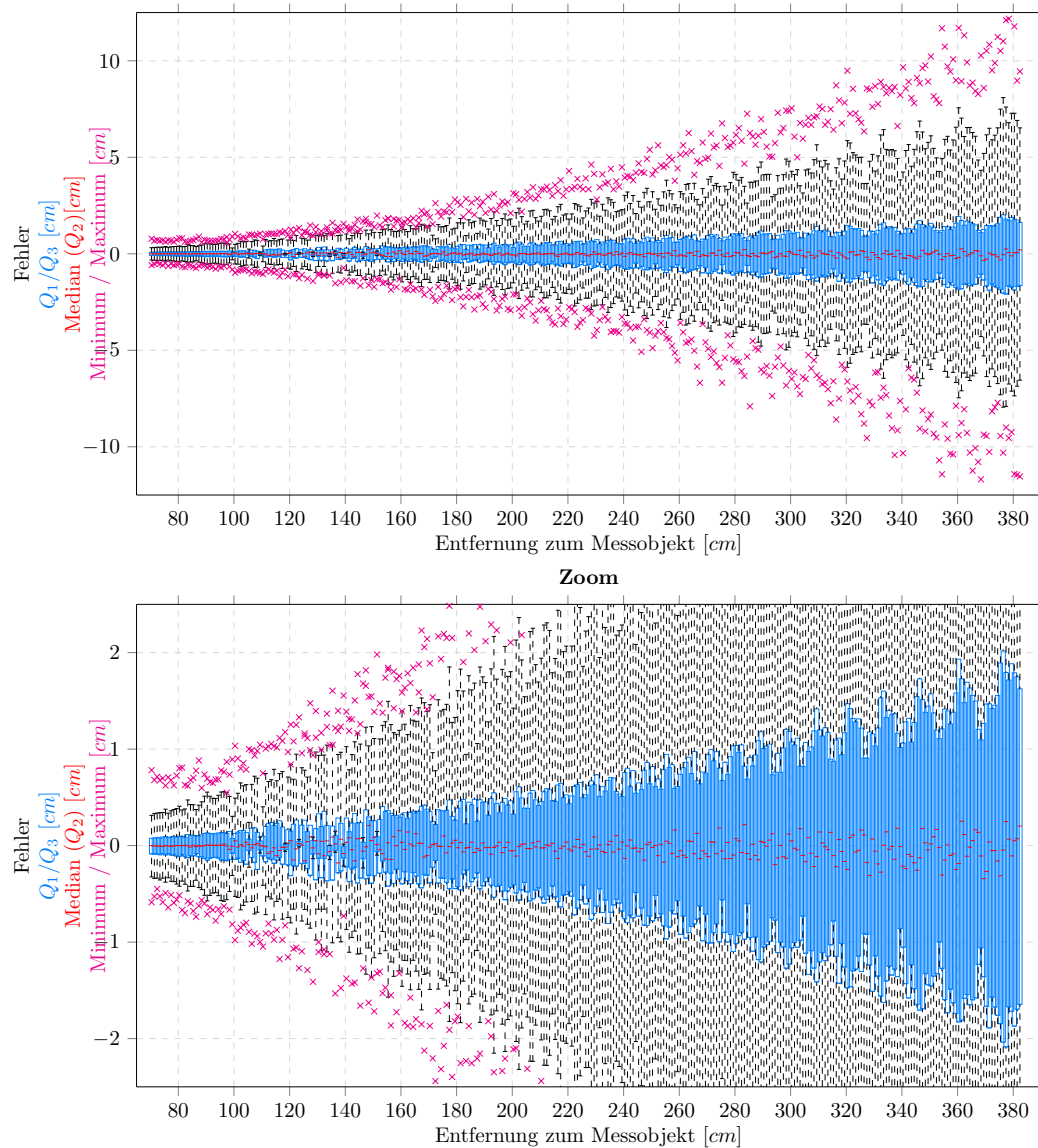


Abbildung 4.1.: Boxplots der Fehler der Tiefenkamera in unterschiedlichen Entfernungen

angegebenen Entfernung auftritt.

Bis zu einer Entfernung von 1,5 m kann herausgelesen werden, dass der Fehler von mehr als 99 % der Daten innerhalb von $\pm 1,5$ cm liegt. Der Absolutfehler von 50 % der Daten sogar unterhalb von 0,5 cm ($\pm 0,5$ cm). Bis zu einer Entfernung von 3 m liegt der Fehler für 50 % der Daten bei etwa $\pm 1,25$ cm und für 90 % bei ± 5 cm, welches

4.2. Ergebnisse zur Messgenauigkeit der Tiefenkamera

Tabelle 4.2.: Fehlerbereiche in der Übersicht

Entfernung	50% < IQA ($Q_3 - Q_1$)	99% < 4 IQA (Antennen)	max.
1.0 m	0.3 cm	1.1 cm	2.0 cm
1.5 m	0.7 cm	2.6 cm	3.2 cm
2.0 m	1.1 cm	4.3 cm	5.5 cm
2.5 m	1.7 cm	6.8 cm	9.5 cm
3.0 m	2.5 cm	9.9 cm	14.2 cm
3.5 m	3.4 cm	13.7 cm	20.0 cm
3.8 m	4.1 cm	16.0 cm	23.2 cm

einem Absolutfehler unterhalb von 1,7 % prozentual zur Entfernung entspricht.

4. Ergebnisse

Tiefenmessung 2

Der zur Erfassung der Messergebnisse verwendete Ablauf ist im Teilbereich *Tiefenmessung 2* aus Kapitel 3.7.1 beschrieben. Pro Entfernung (Schrittweite 20 cm von $52,4\text{ cm} - 392,4\text{ cm}$) wurden 5 Objekte mit jeweils 5 Aufnahmen zur Auswertung verwendet. Auf dem Schachbrett in Abbildung 3.56 sind sechs blaue Linien bzw. acht rote Linien gleicher Länge zu erkennen. Für die Auswertung der blauen, horizontalen Linien in Abbildung 4.2 stehen somit 150 Messwerte pro Entfernung zur Verfügung, für die roten, vertikalen in Abbildung 4.3 entsprechend 200.

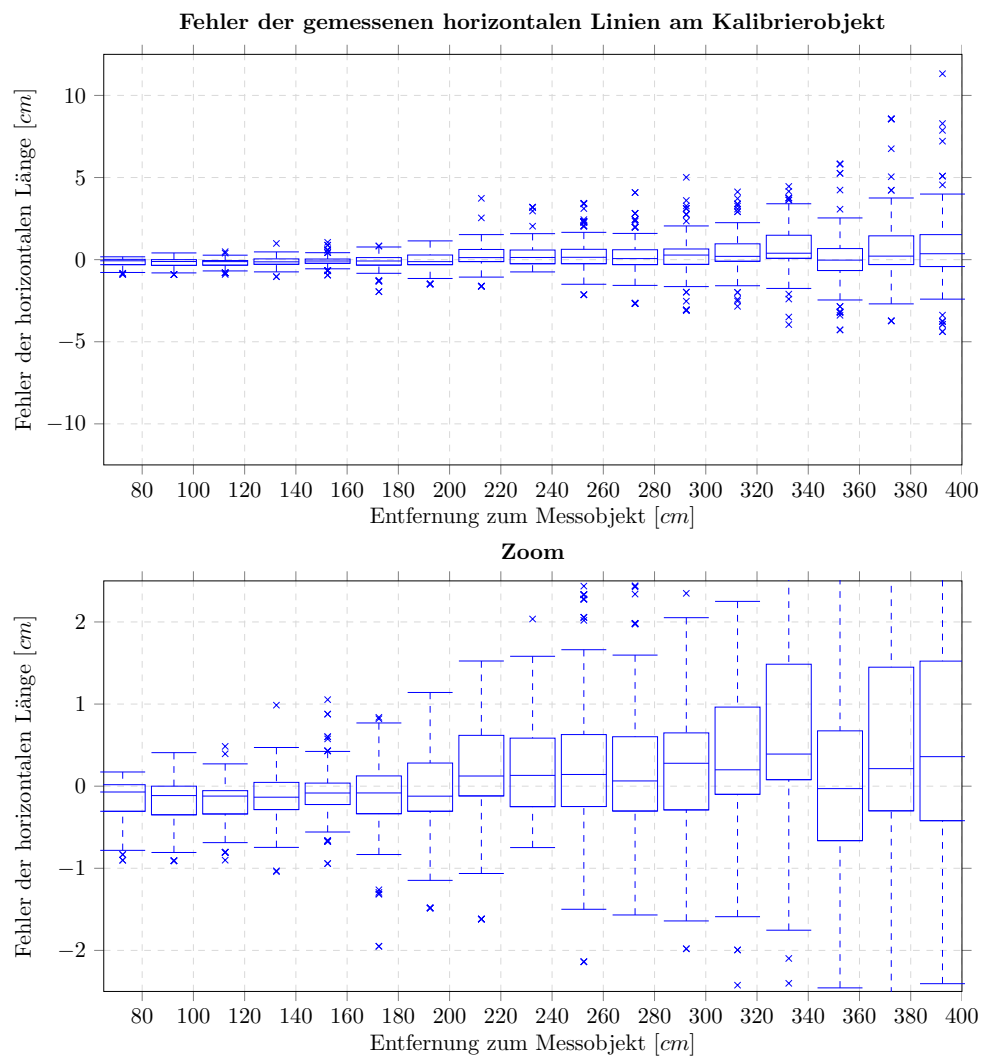


Abbildung 4.2.: Längenfehler der horizontalen Linien (blau) des Kalibrierobjekts in unterschiedlichen Entfernungen

4.2. Ergebnisse zur Messgenauigkeit der Tiefenkamera

Es ist zu beachten, dass lediglich eine im Vergleich zur vorherigen Untersuchung (Unterkapitel *Tiefenmessung 1*) geringe Messwertanzahl (150 bzw. 200 Messwerte) zur Verfügung steht. Der Fehler der Länge (Längendifferenz) ist, wie zuvor im Unterkapitel *Tiefenmessung 1*, auf der Ordinate dargestellt in Boxplots (siehe Kapitel 3.8.1) gegen die Entfernung des Kalibrierobjekts zur Kamera auf der Abszisse aufgetragen.

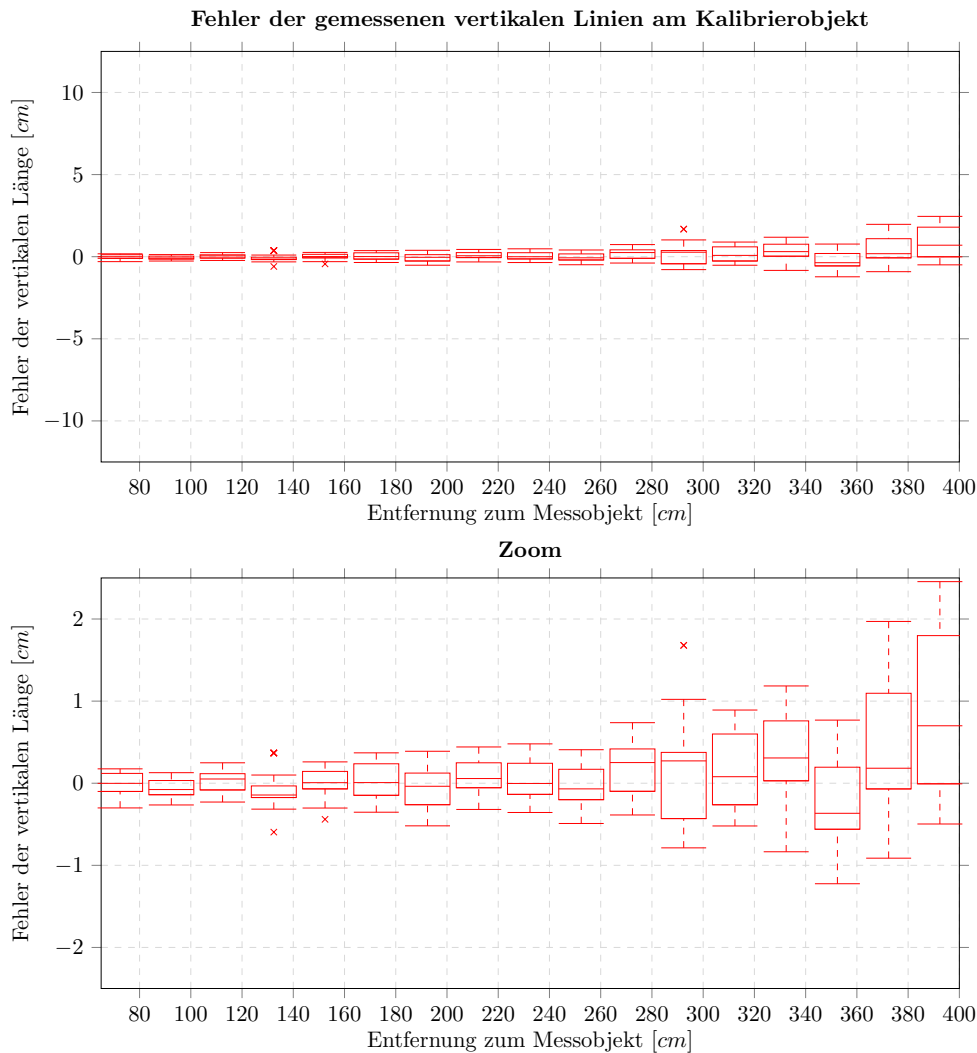


Abbildung 4.3.: Längenfehler der vertikalen Linien (rot) des Kalibrierobjekts in unterschiedlichen Entfernungen

Die Mediane der Linien in vertikaler Richtung in Abbildung 4.3 entfernen sich ein wenig mehr von der Null Position, als die horizontalen in Abbildung 4.2. Bei genauerer Betrachtung der Median-Positionen innerhalb der Boxplots, weisen die Daten der horizontalen Linien eine minimal größere Schiefe auf als die Daten der vertikalen

4. Ergebnisse

Linien oder der Auswertung des vorherigen Unterkapitels *Tiefenmessung 1*.

Für die Daten der horizontalen Linien aus Abbildung 4.2 wurden einige Fehlerbereiche, entsprechend der Tabelle 4.2 des vorherigen Unterkapitels, in Tabelle 4.3 zusammengefasst. Die Fehlerbereiche der Daten aus Abbildung 4.3 zu den vertikalen Linien sind in Tabelle 4.4 zusammengefasst.

Durchgehend erkennbar ist, dass die Fehler der Längenbestimmung der vertikalen als auch der horizontalen Linien geringer bzw. gleich (Ausnahme 1 *m* in der dritten Spalte von Tabelle 4.3) der zuvor dargestellten Messungen des vorherigen Unterkapitels 4.2 ausfallen.

Tabelle 4.3.: Horizontale Längenfehler in der Übersicht

Entfernung	50% < IQA ($Q_3 - Q_1$)	85% < 4 IQA (Antennen)	max.
1.0 <i>m</i>	0.3 <i>cm</i>	1.2 <i>cm</i>	1.3 <i>cm</i>
1.5 <i>m</i>	0.3 <i>cm</i>	1.2 <i>cm</i>	2.0 <i>cm</i>
2.0 <i>m</i>	0.6 <i>cm</i>	2.3 <i>cm</i>	2.8 <i>cm</i>
2.5 <i>m</i>	0.9 <i>cm</i>	3.2 <i>cm</i>	5.6 <i>cm</i>
3.0 <i>m</i>	0.9 <i>cm</i>	3.7 <i>cm</i>	8.1 <i>cm</i>
3.5 <i>m</i>	1.4 <i>cm</i>	5.2 <i>cm</i>	10.1 <i>cm</i>
3.9 <i>m</i>	1.9 <i>cm</i>	6.4 <i>cm</i>	15.7 <i>cm</i>

Tabelle 4.4.: Vertikale Längenfehler in der Übersicht

Entfernung	50% < IQA ($Q_3 - Q_1$)	92% < 4 IQA (Antennen)	max.
1.0 <i>m</i>	0.2 <i>cm</i>	0.5 <i>cm</i>	0.5 <i>cm</i>
1.5 <i>m</i>	0.2 <i>cm</i>	0.6 <i>cm</i>	1.0 <i>cm</i>
2.0 <i>m</i>	0.4 <i>cm</i>	0.9 <i>cm</i>	1.0 <i>cm</i>
2.5 <i>m</i>	0.4 <i>cm</i>	0.9 <i>cm</i>	1.0 <i>cm</i>
3.0 <i>m</i>	0.8 <i>cm</i>	1.8 <i>cm</i>	2.5 <i>cm</i>
3.5 <i>m</i>	0.9 <i>cm</i>	2.0 <i>cm</i>	2.5 <i>cm</i>
3.9 <i>m</i>	1.8 <i>cm</i>	3.0 <i>cm</i>	3.0 <i>cm</i>

Zudem sind die Fehler der vertikalen Linien in Tabelle 4.4 durchgehend kleiner (bzw. kleiner gleich) der Fehler, welche für die horizontalen Linien (Tabelle 4.3) gemessen wurden. Eine Interpretation der festgestellten Unterschiede findet sich in Kapitel 5.1.2

4.3. Orientierung der Pferdeköpfe

Die Orientierung der Pferdeköpfe aus Kapitel 3.7.3 im Raum wurde nach dem in Kapitel 3.7.4 beschriebenen Verfahren bestimmt. Die Winkel der Daten aus Kapitel 3.7.3 sind in Abbildung 4.4 in einem Histogramm aufgetragen. Die Abbildung 4.4a zeigt die 2D Draufsicht des Histogramms. Die jeweilige Helligkeit gibt die Anzahl pro Winkelkombination der Winkel ψ und θ wieder. Umso dunkler das Blau, desto größer die Anzahl. Die Schrittweite mit der die Winkel von -45° bis 45° aufgetragen sind beträgt jeweils $2,5^\circ$, wodurch sich das Histogramm aus 36 Winkelbereichen pro Winkel zusammensetzt.

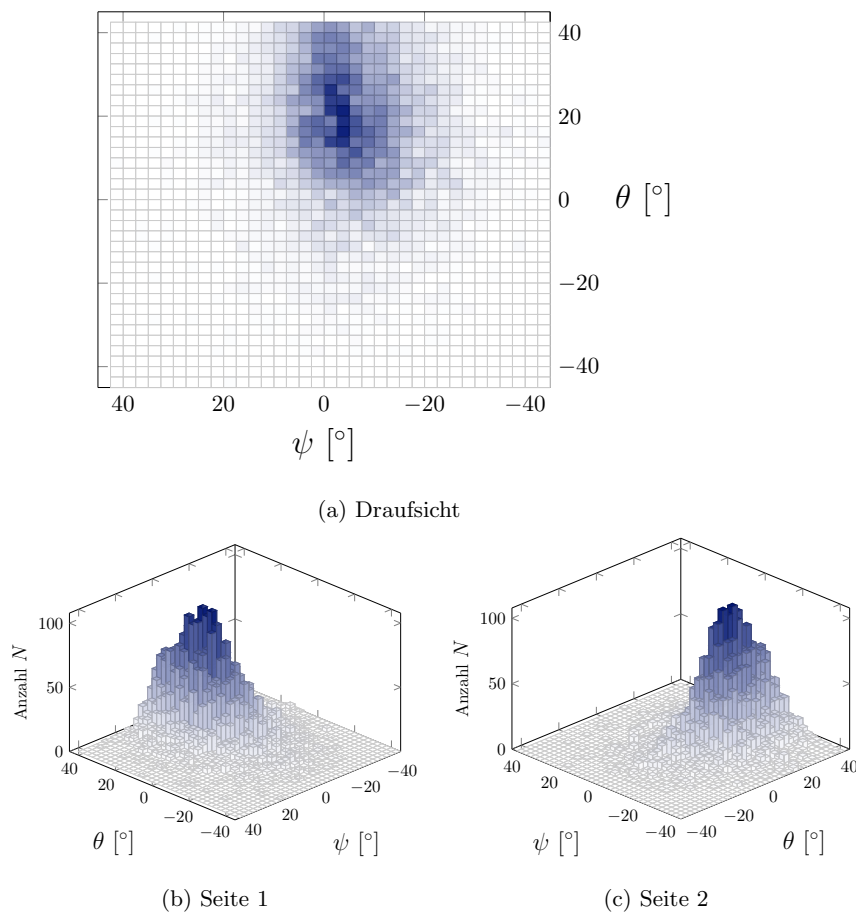


Abbildung 4.4.: Histogramm über die Orientierungswinkel der *Vergleichsdaten*

In den Abbildungen 4.4b/c ist das Histogramm als 3D Grafik dargestellt, wobei zwei Ansichten gewählt wurden um verdeckte Bereiche in der einen Abbildung (4.4b) in der zweiten (4.4c) erkennen zu können. Die Höhe des jeweiligen Balkens im

4. Ergebnisse

Histogramm entspricht der Anzahl der in dem entsprechenden Winkelbereich erfassten Datensätze. Die Helligkeit des Blautons, welche sich mit der Anzahl verändert, wurde für die Balken aus der 2D Aufsicht übernommen.

Berücksichtigt wurden die Daten, für die, nach der Beschreibung in Kapitel 3.7.3, der komplette Satz an Markern gesetzt wurde (Augen und Nüstern) und die transformierten Marker des jeweiligen Modells eines Tieres innerhalb des beschriebenen Rahmens liegen. Die Gesamtzahl der Datensätze, welche zur Erstellung der Abbildung 4.4 verwendet wurde, beträgt 10.207. Von diesen Datensätzen sind 2.517 nach Kapitel 3.7.4 als Tierköpfe mit Selbstverdeckung (entsprechend der Beispiele aus Abbildung 3.64) markiert. Die restlichen 7.690 Datensätze enthalten entsprechende Bilddaten, in denen keine oder nur geringe Selbstverdeckung auftritt. Abbildung 4.5 stellt die beiden Gruppierungen (mit und ohne Selbstverdeckung) mit unterschiedlichen Farben in einem Histogramm dar.

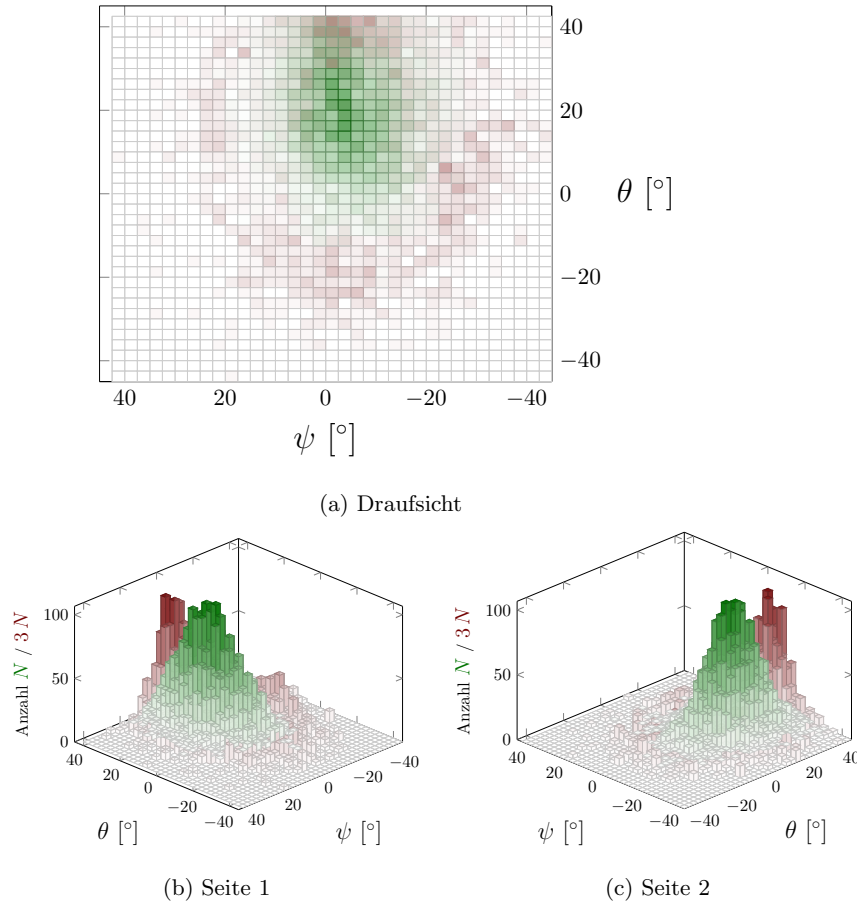


Abbildung 4.5.: Histogramm über die Orientierungswinkel der *Vergleichsdaten* mit und ohne Selbstverdeckung

4.3. Orientierung der Pferdeköpfe

Die Daten mit Selbstverdeckung sind dabei in Rot gekennzeichnet, die ohne Selbstverdeckung in Grün. Durch die geringere Anzahl der Datensätze mit Selbstverdeckung und die größere Fläche, auf die sich die Daten im Histogramm verteilen, wurde die Anzahl auf das dreifache ($3N$) skaliert. Dies soll die Daten in Abbildung 4.5 besser sichtbar machen.

In Abbildung 4.6a ist eine rote Ellipse über die 2D Draufsicht des Histogramms aus Abbildung 4.5a gelegt. Die Orientierungswinkel innerhalb dieser Ellipse können als Bereich angenommen werden, in dem eine Selbstverdeckung lediglich vereinzelt auftritt, so dass Bilddaten, die keine Selbstverdeckung haben sollten, innerhalb der in Grün dargestellten Ellipse aus Abbildung 4.6b liegen sollten. Der rote Bereich aus

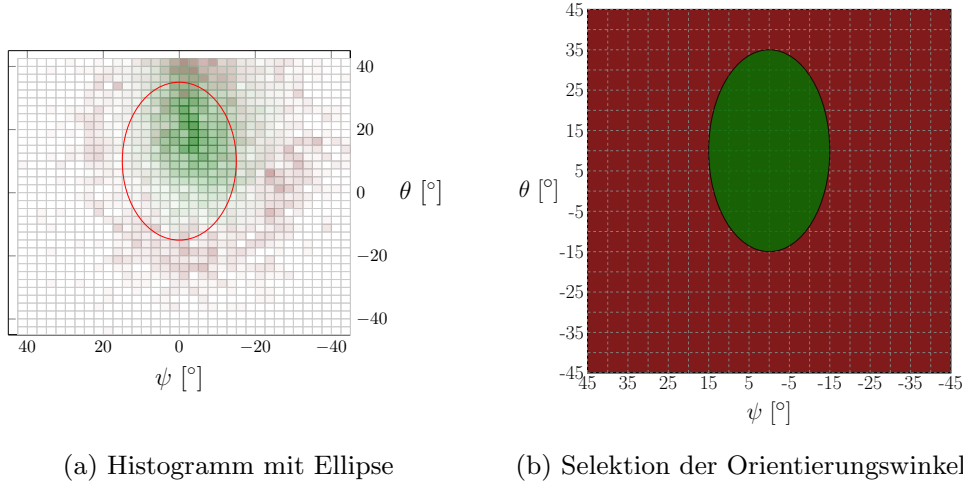


Abbildung 4.6.: Orientierungswinkel zur Bestimmung von Selbstverdeckung

Abbildung 4.6b stellt entsprechend den Bereich der Orientierungswinkel dar, in dem vermehrt mit Selbstverdeckung zu rechnen ist. Die dargestellte Ellipse hat ihr Zentrum in der Winkelposition $\theta = 10^\circ, \psi = 0^\circ$ und hat einen Radius von 25° entlang des Winkels θ bzw. 15° entlang des Winkels ψ .

Mit der Bestimmung der Orientierungswinkel der Pferdeköpfe und dem durch die Ellipse angegebenen Bereich, kann ein Ausschluss von durch Selbstverdeckung des Kopfes zur Erkennung ungünstigen Daten erfolgen.

Die in Abbildung 4.6 dargestellte Selektion der Orientierungswinkel lässt sich als ein Klassifikator der zuvor den Klassen **p** und **n** (ohne Selbstverdeckung / mit Selbstverdeckung) zugeordneten Daten auffassen. Mit der Prognose (Winkel innerhalb $\Rightarrow \mathbf{p}'$ bzw. außerhalb $\Rightarrow \mathbf{n}'$ der Ellipse) lässt sich das Resultat in einer Wahrheitsmatrix entsprechend Tabelle 4.5, wie sie in Kapitel 3.8.2 vorgestellt wird, eintragen:

4. Ergebnisse

Tabelle 4.5.: Wahrheitsmatrix nach Selektion mittels Winkelerorientierung

	p	n	
p'	5467	520	5987
n'	2223	1997	4220
	7690	2517	10207

Aus dieser lässt sich eine Treffergenauigkeit (*engl.: accuracy*) von 73% nach Gleichung 3.129 für die Selektion über die Orientierungswinkel der Pferdeköpfe mit der vorgegebenen Ellipse bestimmen:

$$\frac{TP + TN}{P + N} = \frac{5467 + 1997}{7690 + 2517} = 0,73. \quad (4.1)$$

4.4. Ergebnisse der Detektion

4.4.1. Parametervergleich

Parametervergleich I

Der Algorithmus aus Kapitel 3.4.2 besitzt nach Tabelle 3.5 (Kapitel 3.7.5) 9 Parameter. In Tabelle 4.6 sind die für die 15 Lerndatensätzen (siehe *Parameteroptimierung I* aus Kapitel 3.7.5) optimierten Parameter in den Spalten 2 – 10 aufgeführt. Zudem sind die Gesamtkosten c_I nach Gleichung 3.111 jeweils für den Lerndatensatz (Spalte 11 der Tabelle 4.6), als auch für den Testdatensatz angegeben (\hat{c}_I in der 12. Spalte von Tabelle 4.6).

Tabelle 4.6.: Parameter zum Clustern aus Kapitel 3.4.2

#	s_ψ	s_θ	s_δ	p_ψ	p_θ	p_δ	T_p	N_{min}	N_{Rmax}	c_I	\hat{c}_I
1	26,5°	1,1°	2,1 cm	52,9°	32,6°	30,0 cm	1,6 %	49	29	102,6	109,7
2	5,0°	46,0°	2,0 cm	59,7°	46,7°	24,1 cm	58,2 %	11	189	128,9	132,9
3	38,5°	3,0°	2,8 cm	53,2°	30,1°	30,0 cm	0,0 %	130	173	108,0	106,9
4	5,0°	48,9°	2,0 cm	50,0°	52,4°	28,0 cm	84,6 %	18	72	128,3	135,7
5	6,8°	47,7°	26,2 cm	60,0°	52,6°	23,8 cm	92,2 %	103	220	115,7	124,3
6	23,9°	51,4°	2,1 cm	59,1°	46,5°	25,2 cm	39,6 %	26	200	123,8	125,7
7	37,4°	4,0°	8,2 cm	49,5°	31,7°	30,0 cm	2,0 %	22	3	108,1	109,9
8	11,6°	24,0°	26,1 cm	54,1°	48,0°	23,1 cm	81,3 %	70	226	122,5	126,9
9	60,0°	3,1°	2,0 cm	51,9°	32,5°	30,0 cm	0,0 %	115	187	105,8	103,2
10	21,2°	9,7°	2,0 cm	60,0°	24,3°	29,4 cm	53,2 %	23	155	122,3	126,2
11	25,7°	1,1°	7,2 cm	51,4°	39,3°	30,0 cm	0,0 %	95	62	109,3	107,3
12	13,6°	1,3°	2,0 cm	54,5°	47,2°	30,0 cm	1,9 %	41	39	109,8	113,8
13	4,9°	48,0°	4,4 cm	49,4°	51,8°	26,2 cm	0,0 %	11	3	116,3	120,2
14	12,1°	2,1°	2,0 cm	52,8°	46,9°	30,0 cm	0,0 %	47	51	111,8	117,1
15	5,0°	46,2°	2,0 cm	60,0°	47,6°	24,0 cm	43,5 %	12	6	118,0	124,7

Auffällig sind die Schwankungen der optimierten Parameter s_ψ , s_θ und s_δ , als auch N_{min} und N_{Rmax} , in Tabelle 4.6 für die einzelnen Datensätze. Die Parameter p_ψ , p_θ und p_δ weisen keine entsprechend hohe Variation zwischen den Datensätzen auf. Trotz der Unterschiede in den Gesamtkosten \hat{c}_I der Testdatensätze zwischen den Datensätzen, sind diese nicht stark verändert zu denen der Lerndatensätze c_I . Ein Zusammenhang zwischen s_θ und den Gesamtkosten c_I bzw. \hat{c}_I kann hier nicht ausreichend geprüft werden, jedoch sind die Gesamtkosten \hat{c}_I stets über 120 ($\hat{c}_I > 120$), sobald $s_\theta > 4^\circ$. Entsprechendes gilt für den Parameter T_p , ist dieser größer 2% ($T_p > 2\%$), ist auch $\hat{c}_I > 120,2$.

Der Algorithmus zur Segmentierung aus Kapitel 3.4.3 besitzt nach Tabelle 3.6 (Kapitel 3.7.5) 3 bis 5 Parameter, wobei die Parameter T_ψ , T_θ und T_δ optional sind und kombiniert einsetzbar sind. Durch die Kombination der Parameter (T_ψ , T_θ und T_δ) ergeben sich insgesamt 7 mögliche Parametersätze:

je 1 Parameter: $\{T_\psi\}$, $\{T_\theta\}$, $\{T_\delta\}$

4. Ergebnisse

je 2 Parameter: $\{T_\psi, T_\theta\}$, $\{T_\psi, T_\delta\}$, $\{T_\theta, T_\delta\}$

alle Parameter: $\{T_\psi, T_\theta, T_\delta\}$

In Tabelle 4.7 sind die Gesamtkosten nach der *Parameteroptimierung I* aus Kapitel 3.7.5 der Lern- c_I bzw. Testdatensätze \hat{c}_I , welche mit der Gleichung 3.111 für jeden der 7 Parameterkombinationen (erste Zeile) bestimmt wurden, aufgeführt. Die erste Spalte der Tabelle 4.7 gibt, wie in Tabelle 4.6, den jeweiligen der 15 Lern- bzw. Testdatensätze an. In Tabelle 4.7 wurde in jeder Zeile, also für jeden Datensatz, das beste Ergebnis zwischen den verwendeten Parametersätzen nach den Gesamtkosten c_I des jeweiligen Lerndatensatzes in Blau und das zweitbeste Ergebnis in Cyan markiert. Entsprechend wurde das beste Ergebnis für die Gesamtkosten \hat{c}_I des jeweiligen Testdatensatzes in Rot und das zweitbeste Ergebnis in Magenta markiert. Dies dient lediglich der Übersicht der Ergebnisse aus Tabelle 4.7.

Tabelle 4.7.: Ergebnisse der Segmentierung aus Kapitel 3.4.3

#	$\{T_\psi\}$		$\{T_\theta\}$		$\{T_\delta\}$		$\{T_\psi, T_\theta\}$		$\{T_\psi, T_\delta\}$		$\{T_\theta, T_\delta\}$		$\{T_\psi, T_\theta, T_\delta\}$	
	c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I
1	105,3	108,8	76,0	76,2	79,1	83,2	75,3	74,9	80,8	83,0	79,1	83,2	75,9	77,7
2	102,3	102,2	72,6	75,8	80,7	80,6	74,1	76,2	80,7	80,6	80,9	80,8	74,8	76,9
3	107,9	110,7	74,8	76,3	79,6	79,1	74,0	73,6	79,6	79,1	79,6	79,1	79,6	79,5
4	105,0	109,6	74,5	75,6	80,6	83,0	74,1	73,4	80,7	82,8	76,6	77,3	76,2	76,0
5	98,7	107,3	73,2	74,9	79,3	81,4	73,5	75,3	79,3	81,4	79,3	81,4	75,9	76,9
6	100,7	100,2	73,8	75,2	77,8	80,3	73,5	74,1	77,0	80,4	74,9	75,3	77,9	80,8
7	106,4	105,3	74,3	77,7	79,7	82,8	73,4	77,3	79,8	82,8	74,8	77,6	74,8	77,8
8	107,2	112,1	76,4	78,9	79,4	87,0	74,0	77,0	79,4	87,0	79,4	87,0	79,4	87,0
9	104,7	106,0	75,3	75,8	83,8	80,1	75,1	74,4	83,7	81,3	84,0	78,3	78,4	76,0
10	99,6	103,3	74,7	77,3	78,3	80,2	73,8	77,7	78,3	79,9	75,6	77,9	78,3	81,2
11	98,0	104,8	74,1	75,6	75,9	83,4	73,3	76,5	75,8	83,2	75,9	83,4	74,0	78,9
12	106,5	105,2	71,8	78,3	75,5	81,9	72,0	76,6	75,3	81,4	73,6	78,1	74,2	77,9
13	106,9	101,7	76,3	74,3	81,2	74,9	75,7	74,5	81,4	75,9	77,4	75,4	78,4	76,4
14	103,5	105,5	75,4	74,2	80,1	78,9	73,4	73,2	81,2	79,4	80,1	78,8	74,6	75,0
15	104,3	110,8	75,2	75,8	80,9	83,1	73,8	75,7	81,8	81,6	75,5	77,1	76,7	77,7

Durch die farbliche Kennzeichnung, lassen sich die beiden Parameterkombinationen $\{T_\theta\}$ und $\{T_\psi, T_\theta\}$ aus Tabelle 4.7 als die Kombinationen mit den besten Gesamtkosten für die Lern- (c_I) als auch die Testdatensätze (\hat{c}_I) identifizieren.

Im Gegensatz dazu ergibt sich für den Parametersatz $\{T_\psi\}$ (Spalten 2/3 in Tabelle 4.7), trotz der optimierten Parameter das schlechteste Ergebnis nach den Gesamtkosten (c_I und \hat{c}_I) zwischen den Parametersätzen (Vergleich zwischen Spalten 2/3 und den restlichen Spalten aus Tabelle 4.7).

Der Parametersatz $\{T_\theta\}$ (Spalten 4/5 in Tabelle 4.7) wurde bereits als einer der zwei besten identifiziert, wobei mehr Datensätze in Cyan (9 von 15) bzw. Magenta (8 von 15) gefärbt sind. Zudem sind 3 bzw. 2 der 15 Datensätze nach den Gesamtkosten c_I der Lerndatensätze bzw. den Gesamtkosten \hat{c}_I der Testdatensätze nicht unter den jeweils zwei Besten.

Das Berücksichtigen allein des Ebenenparameters δ (Parametersatz $\{T_\delta\}$) bei der Segmentierung nach Kapitel 3.4.3 scheint nach den Spalten 6/7 der Tabelle 4.7 im Vergleich zu den restlichen Parameterkombinationen ein durchschnittliches Ergebnis zu erzielen.

Tabelle 4.7 zeigt ab Spalte 8 die erste Parameterkombination $\{T_\psi, T_\theta\}$ aus zwei Parametern. Werden die Ergebnisse der Spalten 4/5 (Parametersatz $\{T_\theta\}$) mit denen der Spalten 8/9 (Parametersatz $\{T_\psi, T_\theta\}$) aus der Tabelle 4.7 verglichen, so ist erkennbar, dass das Berücksichtigen des Ebenenparameters ψ (neben dem Winkel θ) einen Vorteil bei der Segmentierung nach Kapitel 3.4.3 bringt. Für den Parametersatz $\{T_\psi, T_\theta\}$ ist kein Lern- bzw. Testdatensatz zu finden, für den die Gesamtkosten c_I bzw. \hat{c}_I nicht zu den zwei Besten zählen. Zudem sind mehr Datensätze in Blau (12 von 15) und mehr in Rot (10 von 15) gekennzeichnet, so dass deren Ergebnisse zu den Besten des jeweiligen Datensatzes zwischen den Parameterkombinationen zählen.

Die restlichen Parameterkombinationen, deren Ergebnisse ab Spalte 10 in Tabelle 4.7 aufgeführt sind, machen deutlich, dass das Berücksichtigen des Ebenenparameters δ keinen Vorteil gegenüber der Parameterkombination $\{T_\psi, T_\theta\}$ (siehe Spalten 8/9) bringt. Die Ergebnisse der Parametersätze $\{T_\psi, T_\delta\}$, $\{T_\theta, T_\delta\}$ bzw. $\{T_\psi, T_\theta, T_\delta\}$ in Tabelle 4.7 sind bezüglich der entstehenden Gesamtkosten c_I bzw. \hat{c}_I der jeweiligen Datensätze nicht besser als die Ergebnisse der Parameterkombination $\{T_\psi, T_\theta\}$ aus den Spalten 8/9 der Tabelle 4.7.

Im Anhang E.1.1 sind zudem die optimierten Parameter in einzelnen Tabellen entsprechend der Tabelle 4.6 aufgeführt. In den Tabellen E.1-E.7 aus Anhang E.1.1 ist zudem erkennbar, dass die Variation zwischen den Werten der einzelnen Datensätze in den Parametern T_ψ und T_θ höher ist, sobald sich der Parameter T_δ im Parametersatz befindet (Tabellen E.5-E.7).

Aus dem Vergleich der beiden Algorithmen aus Kapitel 3.4.2 und Kapitel 3.4.3 anhand der berechneten Gesamtkosten c_I bzw. \hat{c}_I , kann als vorläufiges Ergebnis die Segmentierung aus Kapitel 3.4.3 als zu bevorzugendes Verfahren eingestuft werden. Nur die Ergebnisse der Parameterkombination $\{T_\psi\}$ (Spalten 2/3) aus Tabelle 4.7 reichen an die Werte (c_I bzw. \hat{c}_I) der Tabelle 4.6 heran, wobei die Parameterkombination $\{T_\psi\}$ (Spalten 2/3) bereits als die schlechteste Parameterwahl der Segmentierung (Kapitel 3.4.3) eingestuft wurde. Weitere Vergleiche der beiden Algorithmen folgen im Unterkapitel *Parametervergleich II* und im Kapitel 4.4.2.

Parametervergleich II

Die Tabellen 4.8 bzw. 4.9 in diesem Kapitel zeigen die Gesamtkosten aus Gleichung 3.112 (Kapitel 3.7.5) für die Lern- c_{II} und die Testdatensätze \hat{c}_{II} . Die 15 Lern- und Testdatensätze sind identisch mit den im Unterkapitel *Parameteroptimierung I* von Kapitel 3.7.5 beschriebenen. Die zuvor durchgeführte Segmentierungen (Schritt I aus Kapitel 3.7.5) mit den Algorithmen aus Kapitel 3.4.2 bzw. 3.4.3 wurden mit den pro Lerndatensatz optimierten Parametern des vorherigen Unterkapitels *Parametervergleich I* durchgeführt. Die Auswertung in Tabelle 4.9 ist neben den festen Parametern $\{\alpha, t_b\}$ auf die optionalen Parameter $\{T_\psi, T_\theta\}$ der Segmentierung aus Kapitel 3.4.3 beschränkt. Der entsprechende Parametersatz $(\{\alpha, t_b, T_\psi, T_\theta\})$ hat sich bereits nach Tabelle 4.7 gegenüber den übrigen als überlegen herausgestellt. Die verbleibenden Tabellen E.20-E.25 sind im Anhang E.1.2 angegeben. Entsprechend der Tabelle 3.7 (*Parameter II* in Kapitel 3.7.5) besitzt der Algorithmus zur Detektion aus Kapitel 3.4.4 3 optionale Parameter $(b_\psi, b_\theta, b_\delta)$. Zur Auswertung kommen die 6 Parameterkombinationen

je 1 Parameter: $\{b_\psi\}, \{b_\theta\}, \{b_\delta\}$

je 2 Parameter: $\{b_\psi, b_\theta\}, \{b_\psi, b_\delta\}, \{b_\theta, b_\delta\}$

zum Einsatz. Die jeweiligen Kombinationen der Parameter zur Detektion sind in der ersten Zeile der Tabellen 4.8 und 4.9 zu finden. Die Gesamtkosten c_{II} und \hat{c}_{II} sind in beiden Tabellen (4.8 und 4.9) nach der *Parameteroptimierung II* (Kapitel 3.7.5) bestimmt und prozentual angegeben (Werte aus Gleichung 3.112 mit 100 multipliziert).

Entsprechend der Tabelle 4.7, des vorherigen Unterkapitels, wurden die nach den Gesamtkosten c_{II} bzw. \hat{c}_{II} zwei besten Einträge der Tabellen 4.8 und 4.9 farblich hervorgehoben. Pro Datensatz sind die Einträge mit den Gesamtkosten der Lerndatensätze c_{II} in Blau bzw. Cyan (kleinster bzw. zweitkleinster Wert) markiert. Die Gesamtkosten der Testdatensätze \hat{c}_{II} sind in Rot bzw. Magenta (kleinster bzw. zweitkleinster Wert) hervorgehoben, sobald sie zu den zwei niedrigsten Einträgen innerhalb einer Zeile der verschiedenen Parameterkombinationen zählen.

Tabelle 4.8 zeigt die Ergebnisse der Detektion von Pferdeköpfen aus Kapitel 3.4.4, welche zum vorherigen Auffinden den Algorithmus aus Kapitel 3.4.2 einsetzt. Durch die farbliche Markierung in der Tabelle 4.8 hebt sich besonders die Parameterkombination $\{b_\psi, b_\delta\}$ gegenüber den übrigen Kombinationen hervor. Die Parameterwahl $\{b_\psi, b_\delta\}$ (Spalten 10/11) ergibt für die Detektion mit vorherigem Clustern nach Kapitel 3.4.2 die besten Ergebnisse nach den Gesamtkosten c_{II} (13/15 blau und 2/15 cyan) bzw. \hat{c}_{II} (12/15 rot und 2/15 magenta). Der Parametersatz $\{b_\psi\}$ (Spalten 2/3) ist bei 14/15 (rot oder magenta) Datensätzen nach den Gesamtkosten \hat{c}_{II} der Testdaten und bei 8/15 Datensätzen (cyan) nach c_{II} unter den beiden besten Parameterkombinationen.

Tabelle 4.8.: Ergebnisse der Detektion nach dem Clustern aus Kapitel 3.4.2

#	$\{b_\psi\}$		$\{b_\theta\}$		$\{b_\delta\}$		$\{b_\psi, b_\theta\}$		$\{b_\psi, b_\delta\}$		$\{b_\theta, b_\delta\}$	
	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	6,1	9,2	16,7	19,4	12,8	13,8	5,6	9,8	5,8	9,5	9,6	12,2
2	16,7	20,1	28,2	29,0	22,8	26,8	16,7	20,4	12,8	17,2	21,1	25,4
3	8,5	7,9	19,7	18,6	13,3	12,4	8,7	7,9	7,7	7,0	12,3	10,8
4	21,3	27,4	24,3	26,5	22,2	28,8	24,0	25,9	16,0	23,0	19,8	27,0
5	13,2	18,3	24,4	30,9	22,2	29,9	13,7	19,1	10,2	16,6	21,2	28,5
6	20,9	17,9	33,9	36,3	31,0	33,1	21,3	18,6	15,9	17,1	28,8	33,0
7	8,0	8,0	17,1	20,5	13,5	17,0	7,7	9,3	7,8	8,8	8,7	11,1
8	20,5	21,7	30,0	31,2	35,3	45,6	20,6	21,8	17,0	21,4	30,4	42,3
9	8,4	5,5	17,2	16,0	12,5	9,7	8,6	6,0	8,1	5,4	11,3	7,5
10	27,1	24,7	36,2	35,4	32,6	35,4	27,3	24,8	19,6	21,8	29,3	33,8
11	10,8	10,0	19,4	19,8	15,6	18,5	11,4	10,0	9,5	10,2	11,5	14,8
12	9,8	12,6	17,9	19,7	16,7	20,0	9,8	13,4	8,3	12,6	15,5	19,2
13	10,3	13,0	23,5	25,9	13,2	19,0	10,4	13,2	8,0	11,5	12,4	18,2
14	13,5	14,0	23,6	23,6	20,4	25,2	13,3	14,4	10,1	13,0	19,4	20,8
15	13,9	15,7	21,3	28,7	13,8	19,0	14,2	16,0	9,6	13,0	11,8	17,8

Tabelle 4.9 zeigt die Ergebnisse der Detektion von Pferdeköpfen aus Kapitel 3.4.4, bei der die Segmentierung aus Kapitel 3.4.3 mit dem Parametersatz $\{T_\psi, T_\theta\}$ eingesetzt wurde.

Tabelle 4.9.: Ergebnisse der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	$\{b_\psi\}$		$\{b_\theta\}$		$\{b_\delta\}$		$\{b_\psi, b_\theta\}$		$\{b_\psi, b_\delta\}$		$\{b_\theta, b_\delta\}$	
	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	5,9	8,1	4,8	6,4	15,6	19,9	0,8	2,2	1,4	3,0	0,9	2,5
2	6,5	7,5	4,3	6,2	15,9	21,6	0,3	1,5	0,9	3,2	0,7	1,8
3	9,0	8,6	5,2	5,2	16,2	18,9	1,9	2,0	2,7	3,0	1,9	1,9
4	7,5	7,0	4,3	4,5	17,1	17,6	0,8	0,8	1,4	1,2	1,1	1,0
5	9,2	10,4	4,6	5,8	23,7	23,4	0,7	2,2	1,3	3,3	1,1	2,9
6	9,9	10,3	4,9	12,9	22,3	21,9	1,7	1,7	2,5	2,4	2,0	1,9
7	9,5	8,4	3,8	6,5	19,0	20,9	2,0	1,5	3,3	3,9	2,2	1,8
8	7,4	9,9	5,5	5,8	15,4	15,9	1,7	2,2	1,8	3,1	1,5	2,2
9	7,3	7,2	4,4	5,3	17,5	16,5	1,2	1,5	1,5	1,7	1,3	1,5
10	7,4	11,4	5,2	6,3	19,8	21,4	1,1	2,5	2,3	4,7	2,0	2,5
11	8,1	8,7	4,9	5,5	23,1	26,1	1,0	2,3	1,6	2,9	1,1	2,3
12	7,4	7,1	4,4	5,4	16,0	17,0	0,9	2,0	0,8	2,2	0,8	2,0
13	7,1	7,8	5,4	5,2	16,2	17,9	0,7	1,7	1,1	2,4	1,0	2,5
14	8,8	8,9	5,9	4,3	16,7	17,2	1,6	1,0	2,3	1,7	1,3	1,1
15	7,9	6,9	5,0	6,0	19,1	16,5	0,5	1,3	1,3	1,5	1,0	1,1

In Tabelle 4.9 hebt sich der Parametersatz $\{b_\psi, b_\theta\}$ (Spalten 8/9) gegenüber den restlichen Parameterkombinationen als der nach den Gesamtkosten c_{II} bzw. \hat{c}_{II} Beste hervor (10/15 rot bzw. 15/15 rot/magenta und 12/15 blau bzw. 14/15 blau/cyan). Gefolgt von der Parameterkombination $\{b_\theta, b_\delta\}$ (Spalten 12/13), welche im Vergleich der zweitbeste Parametersatz ist (12/15 cyan bzw. 15/15 blau/cyan und 9/15 magenta bzw. 14/15 rot/magenta). Der Parametersatz $\{b_\theta\}$ (Spalten 4/5) schneidet nach

4. Ergebnisse

den Gesamtkosten c_{II} bzw. \hat{c}_{II} im Vergleich zu den Parametersätzen $\{b_\psi\}$ (Spalten 2/3) und $\{b_\delta\}$ (Spalten 6/7) besser ab. Außerdem gehören $\{b_\psi, b_\theta\}$ und $\{b_\theta, b_\delta\}$, wie bereits beschrieben, zu den beiden besten Parametersätzen. Dies lässt den Schluss zu, dass der Parameter b_θ als der wichtigste Parameter der insgesamt drei optionalen Parametern (Tabelle 3.7 aus Kapitel 3.7.5) zur Detektion nach der Segmentierung aus Kapitel 3.4.3 ist.

Ein entsprechender Schluss lässt sich aus Tabelle 4.8 für die Detektion nach dem Clustern aus Kapitel 3.4.2 nicht eindeutig ziehen.

Beim Vergleich der Ergebnisse der unterschiedlichen Algorithmen aus den Tabellen 4.8 und 4.9, wird ersichtlich, dass Tabelle 4.9 die deutlich geringeren Gesamtkosten c_{II} bzw. \hat{c}_{II} enthält. Lediglich der Parametersatz $\{b_\delta\}$ (Spalte 6/7) aus Tabelle 4.9 liefert vergleichbare Ergebnisse zu denen in Tabelle 4.8. Nur in Einzelfällen kommen die Ergebnisse aus Tabelle 4.8 an die des Parametersatzes $\{b_\psi\}$ (Spalten 2/3) aus Tabelle 4.9 heran, wobei die Ergebnisse des Parametersatzes $\{b_\psi\}$ aus Tabelle 4.9 für keinen Datensatz zu den zwei besten gehören. Dadurch stellt sich, den Ergebnissen dieses Kapitels zufolge, wiederholt der Algorithmus zur Segmentierung aus Kapitel 3.4.3 gegenüber dem Clustern nach Kapitel 3.4.2 auch in Kombination mit der Detektion (siehe Schritt *I + II* aus Kapitel 3.7.5) als überlegen dar.

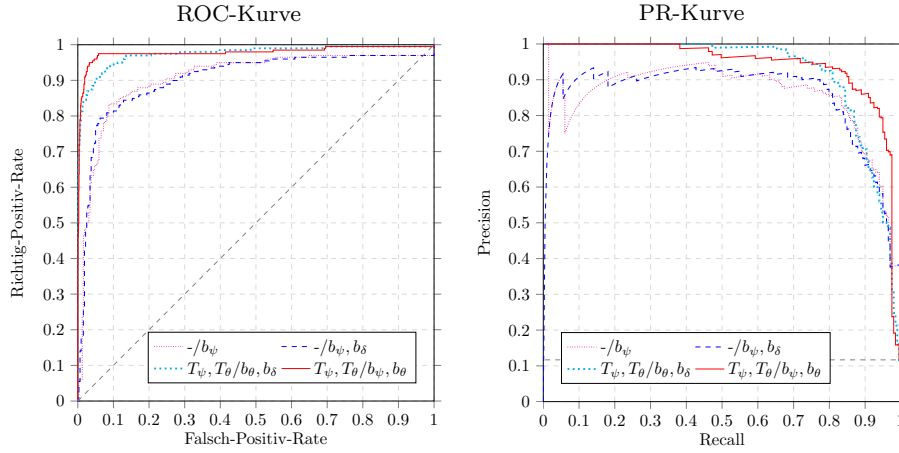
4.4.2. Detektionsraten

Die Gesamtkosten \hat{c}_{II} des Schritt *II* aus Kapitel 3.7.5 (*Parameteroptimierung II*) basieren auf der Berechnung der Fläche unter der ROC-Kurve (AUC). Die Gesamtkosten \hat{c}_{II} entsprechen dabei der Fläche, welche nicht unter der ROC-Kurve liegt ($1 - AUC$, siehe Gleichung 3.112). Die Abbildungen 4.7-4.12 stellen die zwei, im Kapitel 4.4.1 beschriebenen, besten Parameterkombinationen (nach den Gesamtkosten \hat{c}_{II}) für jeden der beiden Algorithmen aus Kapitel 3.4.2 und Kapitel 3.4.3 mit den gesamten ROC-Kurven dar. Zur Vollständigkeit ist zu jeder ROC-Kurve die entsprechende PR-Kurve (Precision/Recall-Kurve siehe Kapitel 3.8.2) angegeben.

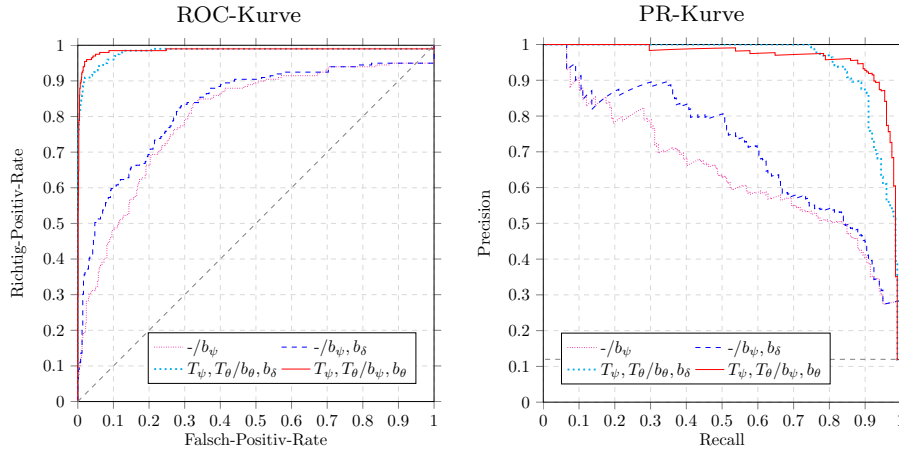
In den Abbildungen 4.7-4.12 ist die Detektion der Pferdeköpfe aus Kapitel 3.4.4 (Schritt *I*) nach der Segmentierung aus Kapitel 3.4.3 (Schritt *II*) mit den optionalen Parametern $\{T_\psi, T_\theta\}/\{b_\psi, b_\theta\}$ in Rot (durchgehend) und mit den Parametern $\{T_\psi, T_\theta\}/\{b_\theta, b_\delta\}$ in Cyan (grob gepunktet) dargestellt. Die Detektion (Schritt *I*) mit dem Algorithmus aus Kapitel 3.4.2 ist mit den optionalen Parametern $\{b_\psi, b_\delta\}$ in Blau (gestrichelt) und mit dem Parameter $\{b_\psi\}$ in Magenta (fein gepunktet) dargestellt.

Die rot- bzw. cyanfarbenen ROC- und PR-Kurven liegen oberhalb der blau- bzw. magentafarbenen. Bis auf wenige Ausnahmen liegt zudem die rote ROC-Kurve weiter links oben als die cyan eingefärbten. Dies bestätigt das Ergebnis aus Kapitel 4.4.1, in dem bereits die Segmentierung aus Kapitel 3.4.3 mit den Parametern

$\{T_\psi, T_\theta\}/\{b_\psi, b_\theta\}$ als die dem Clustern nach Kapitel 3.4.2 überlegene Methode hervorgehoben wird. Außerdem bestätigt sich die Überlegenheit der Parameterkombination $\{T_\psi, T_\theta\}/\{b_\psi, b_\theta\}$ gegenüber der Kombination $\{T_\psi, T_\theta\}/\{b_\theta, b_\delta\}$ zur Segmentierung nach Kapitel 3.4.3 für die Mehrzahl der Testdatensätze.



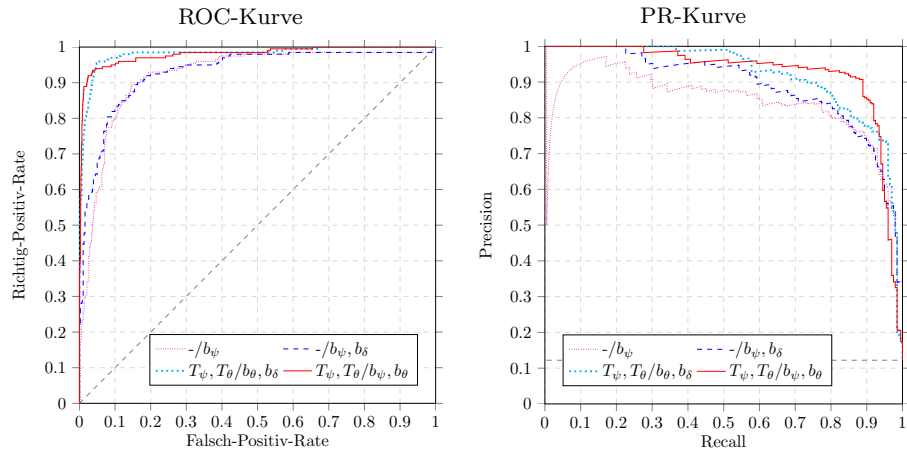
(a) Testdatensatz #1



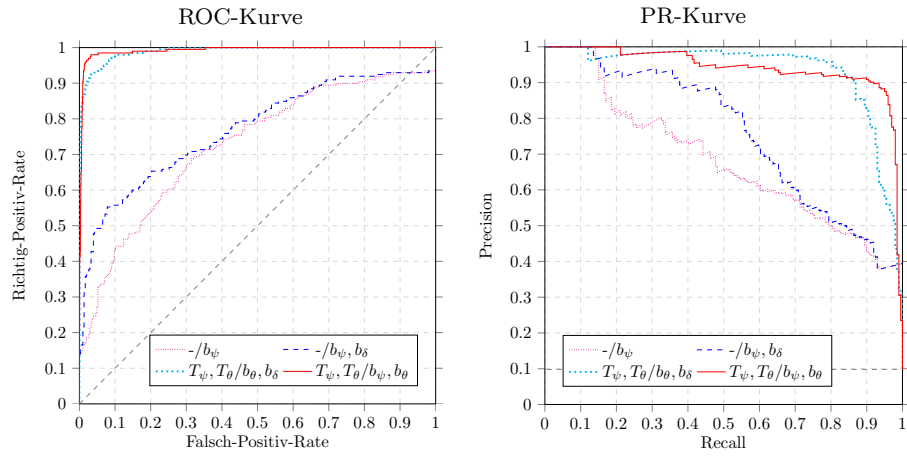
(b) Testdatensatz #2

Abbildung 4.7.: ROC- und PR-Kurven der Testdatensätze 1-2

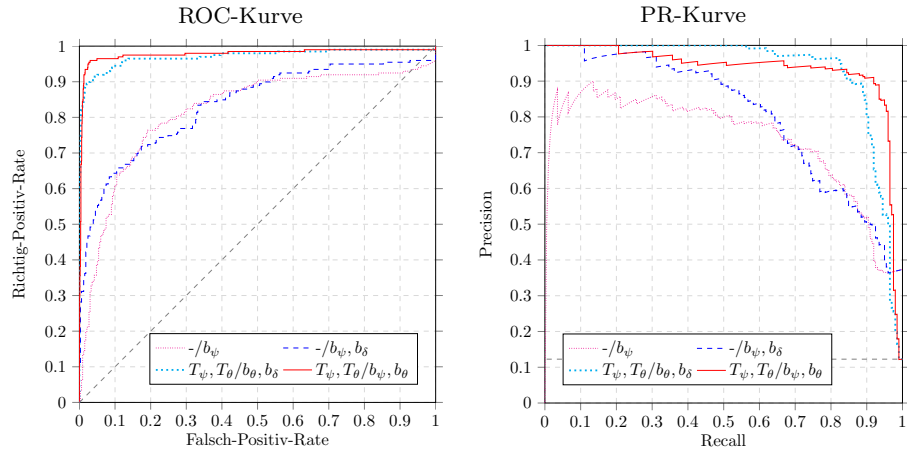
4. Ergebnisse



(a) Testdatensatz #3



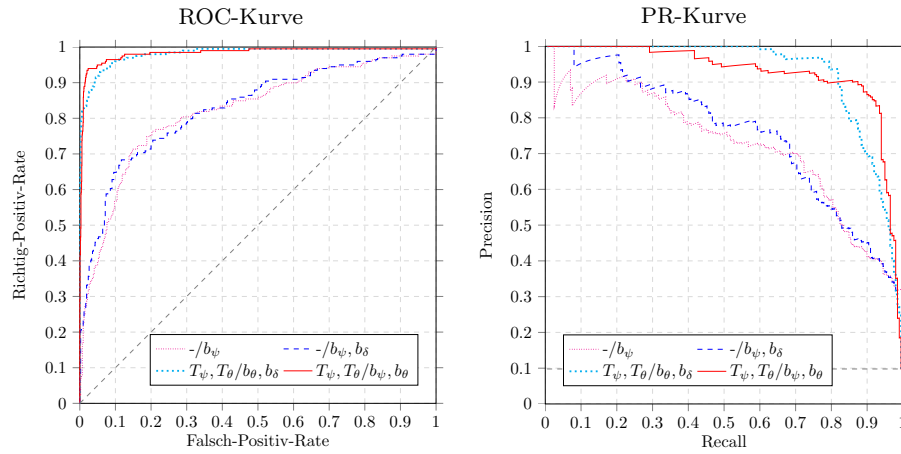
(b) Testdatensatz #4



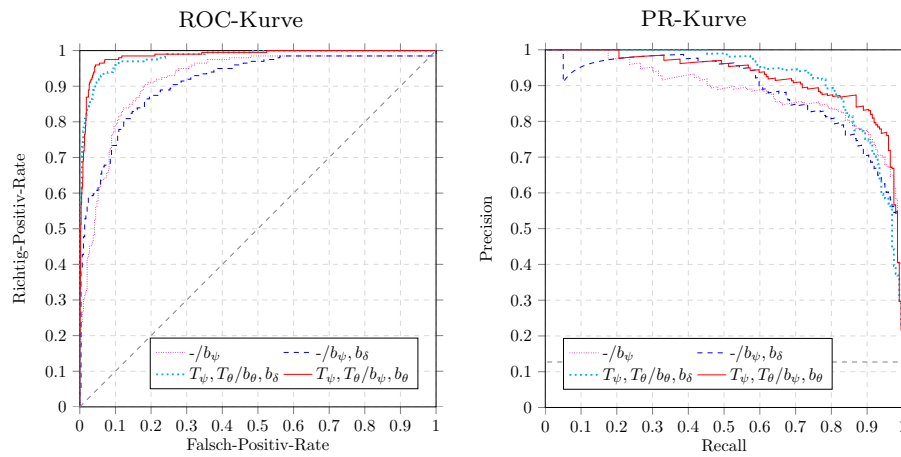
(c) Testdatensatz #5

Abbildung 4.8.: ROC- und PR-Kurven der Testdatensätze 3-5

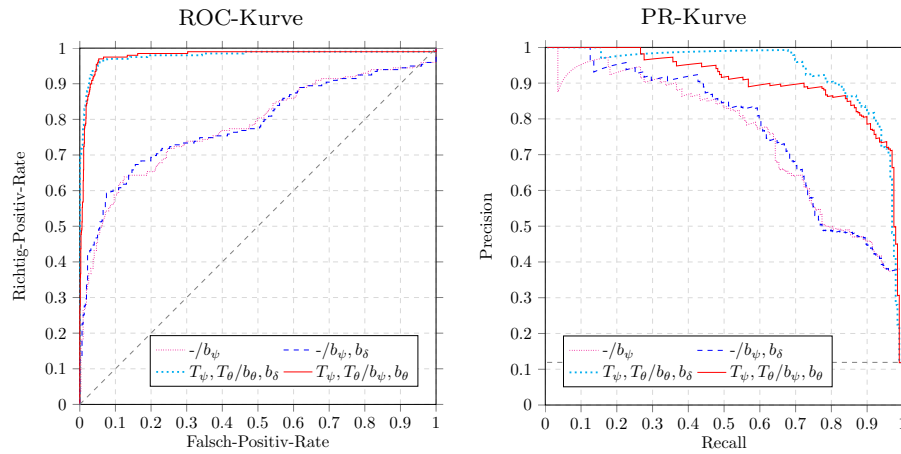
4.4. Ergebnisse der Detektion



(a) Testdatensatz #6



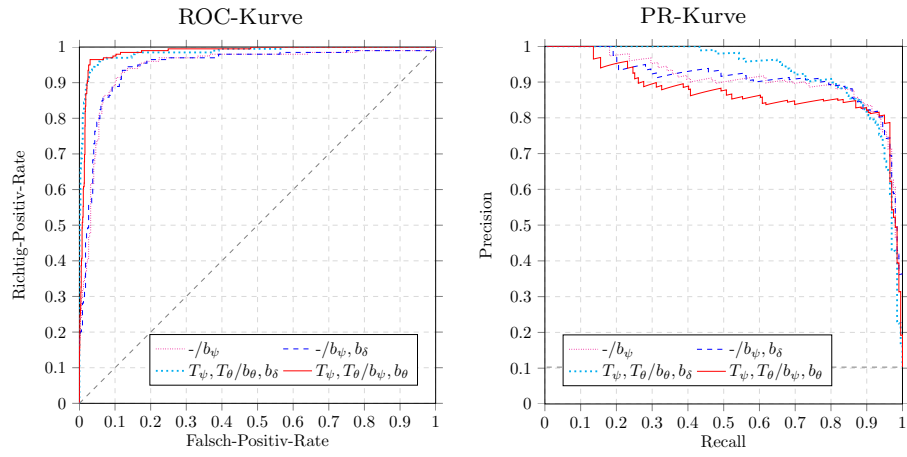
(b) Testdatensatz #7



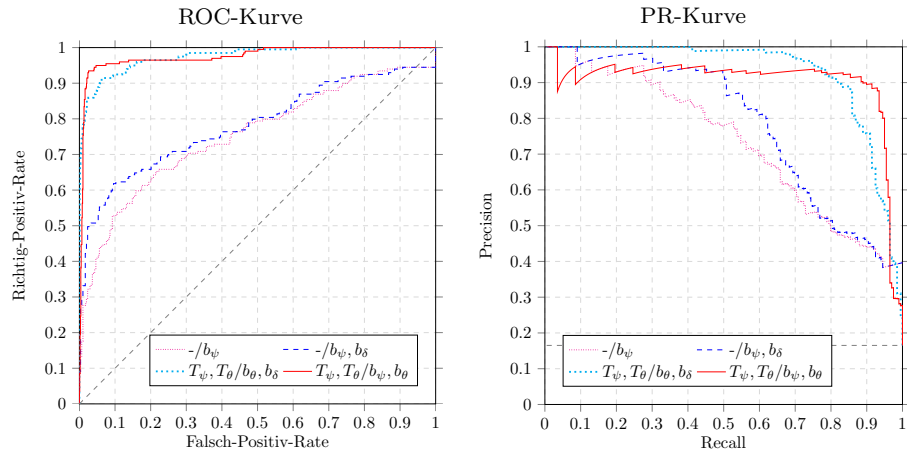
(c) Testdatensatz #8

Abbildung 4.9.: ROC- und PR-Kurven der Testdatensätze 6-8

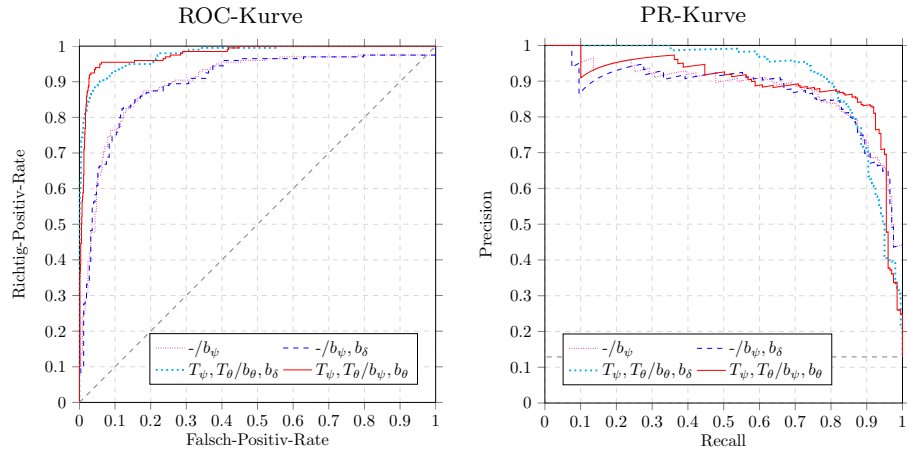
4. Ergebnisse



(a) Testdatensatz #9



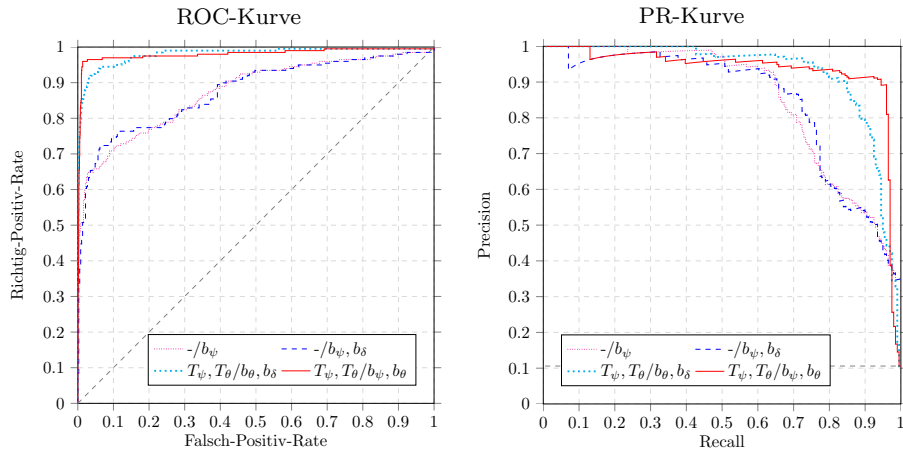
(b) Testdatensatz #10



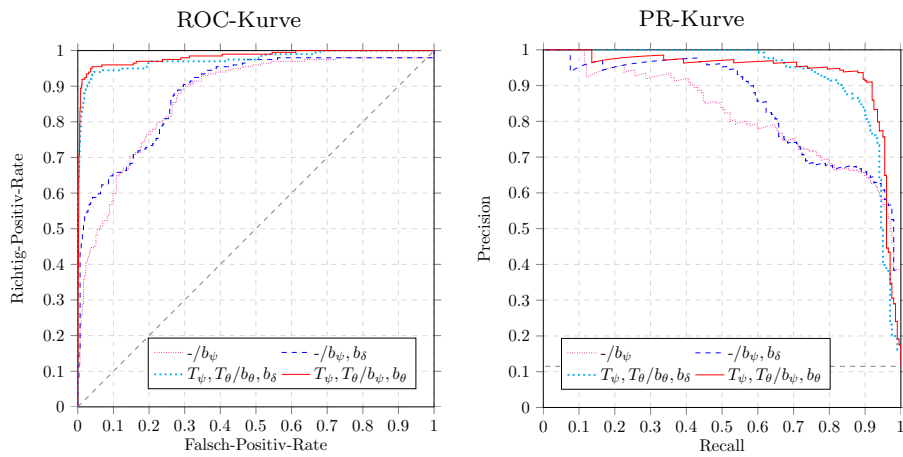
(c) Testdatensatz #11

Abbildung 4.10.: ROC- und PR-Kurven der Testdatensätze 9-11

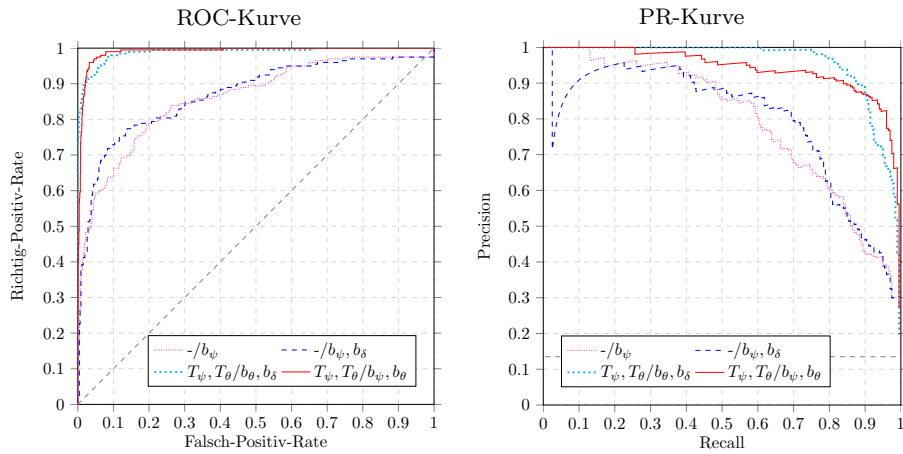
4.4. Ergebnisse der Detektion



(a) Testdatensatz #12



(b) Testdatensatz #13



(c) Testdatensatz #14

Abbildung 4.11.: ROC- und PR-Kurven der Testdatensätze 12-14

4. Ergebnisse

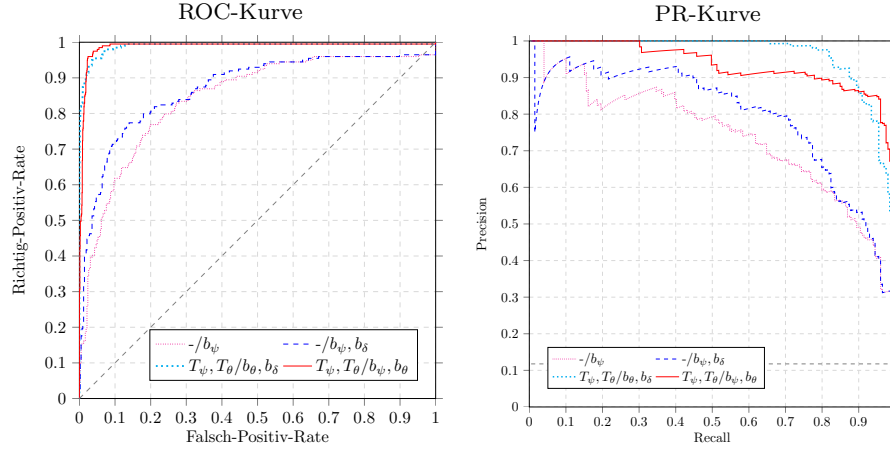


Abbildung 4.12.: ROC- und PR-Kurven des Testdatensatz 15

Die Tabellen 4.10-4.13 fassen, zur Beurteilung der Güte, aus den ROC-Kurven der Abbildungen 4.7-4.12 exportierte Werte zusammen. Die Angaben P und N (Spalten 2/3) entsprechen der Anzahl der Elemente der jeweiligen Klasse (\mathbf{p} bzw. \mathbf{n}). N , die Anzahl der negativen Elemente, ist für den Algorithmus aus Kapitel 3.4.2 (Tabelle 4.12 und 4.13) deutlich kleiner, da der Algorithmus den Hintergrund herausfiltert, in dem beim Algorithmus aus Kapitel 3.4.3 (Tabelle 4.10 und 4.11) die zusätzlichen negativen Elemente liegen. T_{ROC} aus Spalte 4 entspricht dem Schwellwert in der ROC-Kurve, welcher dem Punkt (FPR, TPR) (Spalten 5/6) zugeordnet werden kann. T_{ROC} wurde so gewählt, dass die Treffergenauigkeit ACC in Spalte 8 maximal ist. ACC ist nach Gleichung 3.129 (Kapitel 3.8.2) berechnet. Zusätzlich ist die Genauigkeit PPV (siehe Gleichung 3.128 aus Kapitel 3.8.2) angegeben, wodurch der Punkt (PPV, TPR) in der PR-Kurve angegeben ist. Die Empfindlichkeit TPR (Spalte 6) ist nach Gleichung 3.125 (Kapitel 3.8.2) berechnet, die Ausfallrate FPR (Spalte 5) nach Gleichung 3.126 (Kapitel 3.8.2).

Zu Erkennen ist, dass die mögliche Treffergenauigkeit ACC mit dem Einsatz des Algorithmus aus Kapitel 3.4.3 im Gegensatz zum Algorithmus aus Kapitel 3.4.2 deutlich höher liegt. Die durchschnittliche Treffergenauigkeit aus Tabelle 4.10 liegt bei 97,4 %, die in Tabelle 4.11 bei 97,0 %. Die durchschnittliche Treffergenauigkeit aus Tabelle 4.12 liegt hingegen bei 83,9 %, die in Tabelle 4.13 bei 82,2 %.

Tabelle 4.10.: Treffergenauigkeit ACC der Detektion/Segmentierung (Kapitel 3.4.3)
Parametersatz $\{T_\psi, T_\theta\}/\{b_\psi, b_\theta\}$

#	P	N	T_{ROC}	FPR [%]	TPR [%]	PPV [%]	ACC [%]
1	199	1505	0,19	0,9	84,4	92,3	97,4
2	199	1462	0,23	0,7	89,4	94,7	98,1
3	199	1429	0,17	1,3	88,9	90,8	97,5
4	199	1829	0,16	1,1	93,5	89,9	98,3
5	199	1424	0,15	1,3	92,0	91,0	97,9
6	199	1836	0,13	1,0	87,4	90,2	97,8
7	199	1366	0,22	1,8	86,9	87,4	96,7
8	199	1470	0,20	1,8	83,9	86,5	96,5
9	199	1734	0,17	2,4	93,5	81,6	97,2
10	199	1006	0,18	2,1	91,5	89,7	96,8
11	199	1345	0,17	2,7	91,0	83,4	96,5
12	199	1679	0,15	1,1	94,5	90,8	98,4
13	199	1530	0,16	0,8	89,4	93,7	98,1
14	199	1275	0,23	2,5	93,5	85,3	96,9
15	199	1492	0,26	2,2	95,0	85,1	97,5

Tabelle 4.11.: Treffergenauigkeit ACC der Detektion/Segmentierung (Kapitel 3.4.3)
Parametersatz $\{T_\psi, T_\theta\}/\{b_\theta, b_\delta\}$

#	P	N	T_{ROC}	FPR [%]	TPR [%]	PPV [%]	ACC [%]
1	199	1505	0,20	0,9	80,4	92,5	96,9
2	199	1462	0,21	0,8	84,4	93,9	97,5
3	199	1429	0,22	1,6	79,9	87,4	96,1
4	199	1829	0,21	0,9	86,4	91,5	97,9
5	199	1424	0,16	0,4	81,9	96,4	97,4
6	199	1836	0,19	0,6	81,9	93,7	97,7
7	199	1366	0,20	1,0	78,4	91,8	96,4
8	199	1470	0,20	1,3	82,9	89,7	96,8
9	199	1734	0,20	1,3	84,4	88,4	97,3
10	199	1006	0,22	2,1	85,4	89,0	95,9
11	199	1345	0,19	1,1	78,9	91,3	96,3
12	199	1679	0,20	1,1	84,4	90,3	97,4
13	199	1530	0,19	1,0	81,9	91,6	97,1
14	199	1275	0,20	0,9	84,9	93,9	97,2
15	199	1492	0,20	0,9	87,4	92,6	97,7

4. Ergebnisse

Tabelle 4.12.: Treffergenauigkeit ACC der Detektion/Clustern (Kapitel 3.4.2)
Parametersatz $\{b_\psi, b_\delta\}$

#	P	N	T_{ROC}	FPR [%]	TPR [%]	PPV [%]	ACC [%]
1	199	320	0,06	6,3	79,4	88,8	88,2
2	199	502	0,06	4,8	50,3	80,6	82,5
3	199	381	0,06	7,9	80,4	84,2	88,1
4	199	306	0,08	7,8	55,3	82,1	77,6
5	199	334	0,07	8,4	63,3	81,8	81,1
6	199	420	0,07	9,5	64,8	76,3	82,2
7	199	292	0,24	11,0	77,9	82,9	84,5
8	199	321	0,06	7,5	59,3	83,1	79,8
9	199	348	0,06	6,6	85,9	88,1	90,7
10	199	303	0,07	9,6	61,8	80,9	79,1
11	199	251	0,22	11,6	81,9	84,9	85,6
12	199	370	0,06	5,7	70,9	87,0	86,1
13	199	314	0,06	4,1	58,8	90,0	81,5
14	199	455	0,06	6,6	69,3	82,1	86,1
15	199	423	0,07	8,5	70,4	79,5	84,7

Tabelle 4.13.: Treffergenauigkeit ACC der Detektion/Clustern (Kapitel 3.4.2)
Parametersatz $\{b_\psi\}$

#	P	N	T_{ROC}	FPR [%]	TPR [%]	PPV [%]	ACC [%]
1	199	320	0,20	8,8	82,9	85,5	88,1
2	199	502	0,06	9,6	47,7	66,4	78,3
3	199	381	0,19	7,6	77,4	84,2	87,2
4	199	306	0,06	10,1	44,2	73,9	71,9
5	199	334	0,27	10,5	63,8	78,4	79,9
6	199	420	0,07	13,8	69,3	70,4	80,8
7	199	292	0,23	11,6	82,9	82,9	86,2
8	199	321	0,06	10,9	61,8	77,8	78,7
9	199	348	0,06	6,9	86,4	87,8	90,7
10	199	303	0,16	9,2	52,8	78,9	75,7
11	199	251	0,20	12,4	81,4	83,9	84,9
12	199	370	0,05	2,7	64,3	92,8	85,8
13	199	314	0,21	10,8	64,8	79,1	79,7
14	199	455	0,16	4,4	57,8	85,2	84,1
15	199	423	0,06	9,9	61,8	74,5	81,0

4.4.3. Parameterwahl

Der Parametersatz wird entsprechend der Schritte (I und II) der Optimierung aus Kapitel 3.7.5 in den folgenden Unterkapiteln anhand der jeweiligen Kostenfunktionen (Gleichung 3.111 bzw. 3.112 aus Kapitel 3.7.5) ausgewählt. Die Auswahl beschränkt sich auf die Segmentierung aus Kapitel 3.4.3 (Schritt I), da die Ergebnisse des Algorithmus aus Kapitel 3.4.2 in den Kapiteln 4.4.1 und 4.4.2 trotz individuell angepasster Parameter nicht an die der Segmentierung aus Kapitel 3.4.3 heranreichen.

Parameterwahl I

In Kapitel 3.7.5 wurden 15 Lern- bzw. Testdatensätze beschrieben. Die Parameter des Schritt I der Optimierung nach Gleichung 3.111 (Kapitel 3.7.5) wurden für jeden Lerndatensatz einzeln bestimmt, so dass 15 Parametersätze vorhanden sind. Zur Auswahl wurden aus den Parametersätzen jeweils das Minimum, der Median (siehe Kapitel 3.8.1), das arithmetische Mittel \varnothing (siehe Gleichung 3.115 in Kapitel 3.7.6) und das Maximum berechnet. Die 4 Kombinationen (Minimum, Median, arithmetisches Mittel und Maximum) wurden für die jeweiligen Parameterkombinationen, welche sich durch die optionalen Parameter aus Tabelle 3.6 (Kapitel 3.7.5) ergeben, bestimmt. Anschließend wurde die Kostenfunktion c_I nach Gleichung 3.111 (Kapitel 3.7.5) mit den jeweiligen Parametersätzen für die 15 Lerndatensätze bzw. \hat{c}_I für die 15 Testdatensätze berechnet. Die entsprechenden Ergebnisse sind in den Tabellen E.26-E.28 im Anhang E.2.1 aufgeführt. Wie bereits in Kapitel 4.4.2, wurde das beste Ergebniss (c_I minimal) pro Lerndatensatz in Blau und das zweitbeste in Cyan gekennzeichnet. Entsprechend wurde das beste Ergebniss (\hat{c}_I minimal) pro Testdatensatz in Rot und das zweitbeste in Magenta gekennzeichnet. Anschließend wurde die Anzahl N_{c_I} bzw. $N_{\hat{c}_I}$ bestimmt, die pro Parametersatz angeben wieviele beste bzw. zweitbeste Ergebnisse mit den Lern- bzw. Testdatensätzen im Vergleich zu den restlichen Parametersätzen erzielt werden konnten.

Tabelle 4.14 fasst alle nötigen Daten für die entsprechende Auswertung zusammen, so dass nach der Anzahl N_{c_I} bzw. $N_{\hat{c}_I}$ auf den zu bevorzugenden Parametersatz zu schließen ist. Die erste Spalte fasst nochmals die jeweiligen Parameterkombinationen zusammen, dazu gibt die zweite Spalte an, ob es sich um das Minimum, den Median, das arithmetische Mittel oder das Maximum handelt. In den darauffolgenden Spalten (3 – 5) sind die jeweiligen Parametersätze angegeben. Gehört ein Parameter nicht zum Parametersatz, ist dies durch den Querstrich in dem entsprechenden Feld gekennzeichnet. Die beiden letzten Spalten aus Tabelle 4.14 geben die Anzahl N_{c_I} der besten bzw. zweitbesten Ergebnisse auf den Lerndatensätzen und die Anzahl $N_{\hat{c}_I}$ der besten bzw. zweitbesten Ergebnisse auf den Testdatensätzen an. Die Anzahl der besten und der zweitbesten Ergebnisse ist durch einen Schrägstrich voneinander getrennt. Im Fall, dass keine der Datensätze für den Parametersatz zu den zwei besten Ergebnissen gehören, ist dies durch einen Querstrich gekennzeichnet.

4. Ergebnisse

Tabelle 4.14.: Parameterübersicht der Segmentierung aus Kapitel 3.4.3

		α	t_b	T_ψ	T_θ	T_δ	N_{c_I}	N_{c_I}
$\{\alpha, t_b, T_\psi\}$	Min	46,5°	6,4	3,8°	-	-	-/-	-/-
	Med	48,2°	10,5	6,5°	-	-	-/-	-/-
	\emptyset	48,7°	16,0	6,1°	-	-	-/-	-/-
	Max	52,4°	32,0	8,6°	-	-	-/-	-/-
$\{\alpha, t_b, T_\theta\}$	Min	55,5°	3,1	-	6,1°	-	-/-	-/-
	Med	58,3°	4,4	-	7,2°	-	2/3	2/4
	\emptyset	58,1°	4,5	-	7,5°	-	3/2	4/1
	Max	59,7°	6,3	-	10,1°	-	-/-	-/-
$\{\alpha, t_b, T_\delta\}$	Min	55,9°	13,5	-	-	2,6 cm	-/-	-/-
	Med	58,7°	24,0	-	-	2,8 cm	-/-	-/-
	\emptyset	58,5°	23,1	-	-	2,8 cm	-/-	-/-
	Max	60,3°	32,0	-	-	3,1 cm	-/-	-/-
$\{\alpha, t_b, T_\psi, T_\theta\}$	Min	55,0°	7,4	5,3°	4,5°	-	-/-	-/-
	Med	57,6°	9,3	9,1°	5,6°	-	3/8	5/5
	\emptyset	57,7°	9,1	9,6°	5,6°	-	7/2	4/5
	Max	59,5°	10,5	14,7°	7,0°	-	-/-	-/-
$\{\alpha, t_b, T_\psi, T_\delta\}$	Min	55,9°	13,5	3,3°	-	2,6 cm	-/-	-/-
	Med	59,6°	24,0	15,7°	-	2,9 cm	-/-	-/-
	\emptyset	59,2°	22,6	22,4°	-	3,0 cm	-/-	-/-
	Max	61,0°	32,0	44,7°	-	3,6 cm	-/-	-/-
$\{\alpha, t_b, T_\theta, T_\delta\}$	Min	55,9°	13,4	-	2,9°	2,7 cm	-/-	-/-
	Med	58,0°	24,0	-	32,2°	3,0 cm	-/-	-/-
	\emptyset	58,0°	24,1	-	22,8°	3,7 cm	-/-	-/-
	Max	60,3°	32,0	-	44,9°	5,7 cm	-/-	-/-
$\{\alpha, t_b, T_\psi, T_\theta, T_\delta\}$	Min	55,5°	13,0	3,4°	3,0°	2,7 cm	-/-	-/-
	Med	58,3°	24,0	10,6°	4,3°	3,6 cm	-/-	-/-
	\emptyset	58,1°	22,5	17,0°	10,3°	4,9 cm	-/-	-/-
	Max	60,1°	32,0	43,6°	44,6°	10,9 cm	-/-	-/-

Bereits im Unterkapitel *Parametervergleich I* aus Kapitel 4.4.1 konnte die Parameterkombination $\{\alpha, t_b, T_\psi, T_\theta\}$ als eine der Besten auf den Lern- und den Testdatensätzen identifiziert werden. Die Parameterkombination $\{\alpha, t_b, T_\psi, T_\theta\}$ ist den verbleibenden Kombinationen nach Tabelle 4.14 überlegen. Lediglich der Median und das arithmetische Mittel \emptyset der Parameterkombination $\{\alpha, t_b, T_\theta\}$ zählen bei einigen wenigen Datensätzen zu den zwei besten Ergebnissen. Die Unterschiede in den Ergebnissen der ausgewerteten Kostenfunktion c_I (Gleichung 3.111 aus Kapitel 3.7.5), die in den Tabellen E.26-E.28 aus Anhang E.2.1 aufgeführt werden, sind für den Median bzw. das arithmetische Mittel \emptyset zwischen den Parametersätzen $\{\alpha, t_b, T_\theta\}$ und $\{\alpha, t_b, T_\psi, T_\theta\}$, in denen der Parametersatz $\{\alpha, t_b, T_\theta\}$ bessere Ergebnisse erzielt, kaum relevant.

Somit verbleibt die Wahl zwischen dem Median und dem arithmetischen Mittel \emptyset des Parametersatzes $\{\alpha, t_b, T_\psi, T_\theta\}$. Die Parameter in den Zeilen 14 und 15 der Tabelle 4.14, als auch deren Ergebnisse (Tabellen E.26-E.28 aus Anhang E.2.1), unterscheiden sich nur geringfügig. Wird der Median aus dem Parametersatz $\{\alpha, t_b, T_\psi, T_\theta\}$ in Zeile 14 verwendet, so zählen mehr Lerndatensätze (Summe der Werte in Spalte $N_{c_I} \equiv 11$), als auch mehr Testdatensätze (Summe der Werte in

Spalte $N_{\hat{c}_I} \equiv 10$), zu den beiden besten Ergebnissen als bei Verwendung des arithmetischen Mittels \emptyset aus Zeile 15 ($\sum N_{c_I} = 9, \sum N_{\hat{c}_I} = 9$).

Daher werden zur weiteren Auswertung der Median des Parametersatzes $\{\alpha, t_b, T_\psi, T_\theta\}$ aus Zeile 14 der Tabelle 4.14 verwendet, so dass

$$\alpha = 57,6^\circ, \quad t_b = 9,3, \quad T_\psi = 9,1^\circ \quad \text{und} \quad T_\theta = 5,6^\circ \quad (4.2)$$

festgelegt werden.

Parameterwahl II

Entsprechend der *Parameterwahl I*, soll eine Auswahl der Parameter für den Schritt II zur Detektion der Pferdeköpfe aus Kapitel 3.4.4 getroffen werden. Die optimierten Parametersätze wurden nach Kapitel 3.7.5 berechnet und in Kapitel 4.4.1 verglichen. Für jede Parameterkombination aus Tabelle 3.7 (Kapitel 3.7.5) existieren 15 auf die Lerndatensätze optimierte Parametersätze. Von diesen wurde, wie zuvor bei der *Parameterwahl I*, das Minimum, der Median (siehe Kapitel 3.8.1), das arithmetische Mittel \emptyset (siehe Gleichung 3.115 in Kapitel 3.7.6) und das Maximum berechnet (erste und zweite Spalte in Tabelle 4.15). Die entsprechenden Parameterkombinationen (inkl. Minimum, Median, \emptyset und Maximum) sind in Tabelle 4.15 (Spalten 3 – 5) aufgeführt. Gehört ein Parameter nach Tabelle 3.7 (Kapitel 3.7.5) nicht zum Parametersatz, ist dies durch einen Querstrich gekennzeichnet. Zur Berechnung der Kostenfunktion c_{II} bzw. \hat{c}_{II} nach Gleichung 3.112 (Kapitel 3.7.5) wurde der zuvor in *Parameterwahl I* bestimmte Parametersatz $\{\alpha, t_b, T_\psi, T_\theta\}$ nach Gleichung 4.2 zur Segmentierung verwendet.

Die Ergebnisse der Kostenfunktion (c_{II} bzw. \hat{c}_{II}) sind in den Tabellen E.29-E.31 im Anhang E.2.2 zu finden. In den Tabellen E.29-E.31 wurde wiederum pro Datensatz das beste und zweitbeste Ergebnis der Kostenfunktion c_{II} bzw. \hat{c}_{II} zwischen den Parametersätzen farblich markiert. Entsprechend gibt die Anzahl $N_{c_{II}}$ (vorletzte Spalte in Tabelle 4.15) an, wie häufig der jeweilige Parametersatz das beste bzw. zweitbeste Ergebnis nach der Kostenfunktion c_{II} für die insgesamt 15 Lerndatensätze erreicht hat. Die Anzahl $N_{\hat{c}_{II}}$ (letzte Spalte in Tabelle 4.15) gibt die Häufigkeit der besten bzw. zweitbesten Ergebnisse nach der Kostenfunktion \hat{c}_{II} für die Testdatensätze an. Die Anzahl der besten und zweitbesten Ergebnisse ist durch einen Schrägstrich getrennt dargestellt. Die Fälle, in denen kein Datensatz für den jeweiligen Parametersatz zu den besten bzw. zweitbesten Ergebnissen gehören, wurden durch einen Querstrich gekennzeichnet.

4. Ergebnisse

Tabelle 4.15.: Parameterübersicht der Detektion nach der Segmentierung

Parametersatz (4) der Segmentierung: $\{T_\psi, T_\theta\}$						
		b_ψ	b_θ	b_δ	$N_{c_{II}}$	$N_{c_{\hat{I}}}$
$\{b_\psi\}$	Min	10,9°	-	-	-/-	-/-
	Med	29,7°	-	-	-/-	-/-
	Ø	27,4°	-	-	-/-	-/-
	Max	33,3°	-	-	-/-	-/-
$\{b_\theta\}$	Min	-	0,6°	-	-/-	-/-
	Med	-	16,7°	-	-/-	-/-
	Ø	-	18,3°	-	-/-	-/-
	Max	-	44,9°	-	-/-	-/-
$\{b_\delta\}$	Min	-	-	1,2 cm	-/-	-/-
	Med	-	-	5,1 cm	-/-	-/-
	Ø	-	-	4,9 cm	-/-	-/-
	Max	-	-	9,8 cm	-/-	-/-
$\{b_\psi, b_\theta\}$	Min	7,3°	12,7°	-	1/-	-/-
	Med	13,1°	13,6°	-	-/3	-/2
	Ø	21,3°	13,9°	-	-/3	-/3
	Max	44,9°	14,9°	-	10/1	12/-
$\{b_\psi, b_\delta\}$	Min	0,6°	-	7,5 cm	1/-	-/-
	Med	5,5°	-	11,4 cm	-/-	-/-
	Ø	4,1°	-	11,3 cm	-/-	-/-
	Max	7,3°	-	13,7 cm	-/-	-/-
$\{b_\theta, b_\delta\}$	Min	-	6,4°	11,4 cm	1/1	-/-
	Med	-	14,0°	12,9 cm	1/3	1/4
	Ø	-	13,5°	12,9 cm	1/3	1/4
	Max	-	16,7°	14,5 cm	-/1	1/2

Die Parameterkombination $\{b_\psi, b_\theta\}$ bzw. $\{b_\theta, b_\delta\}$ wurde bereits beim *Parametervergleich II* aus Kapitel 4.4.1 durch die Ergebnissen aus Tabelle 4.9 als die beste bzw. zweitbeste Parameterkombination herausgestellt. Die Ergebnisse aus Tabelle 4.15, in der die Parameter nicht für jeden Datensatz einzeln optimiert sind, entsprechen, ohne eine Betrachtung der vier möglichen Parametersätze (Minimum, Median, Ø und Maximum) einer Parameterkombination, der Beobachtung durch den *Parametervergleich II* aus Kapitel 4.4.1. Die Parameterkombination $\{b_\psi, b_\delta\}$ erzielt nach Tabelle 4.15, im Vergleich zu den restlichen Kombinationen, nur für einige wenige Datensätze beste bzw. zweitbeste Ergebnisse. Die Parameterkombination $\{b_\theta, b_\delta\}$ erzielt für mehr Datensätze das zweitbeste Ergebnis. Die Parameterkombination $\{b_\psi, b_\theta\}$ hebt sich hingegen durch die Anzahl der besten und zweitbesten Ergebnisse von den restlichen Parameterkombination ab.

Die größte Anzahl an besten Ergebnissen für die Lern- und Testdatensätze wird nach Tabelle 4.15 mit dem Parametersatz aus dem Maximum der Parameterkombination $\{b_\psi, b_\theta\}$ erreicht. Das Maximum aus den einzeln optimierten Parametersätzen der Kombination $\{b_\psi, b_\theta\}$ ist damit im Vergleich, nach den Ergebnissen aus Tabelle 4.15, dem Minimum, dem Median und dem arithmetischen Mittel Ø überlegen.

4.4. Ergebnisse der Detektion

Zur weiteren Auswertung wird das Maximum des Parametersatzes $\{b_\psi, b_\theta\}$ aus Zeile 16 der Tabelle 4.15 verwendet, so dass die Parameter

$$b_\psi = 44,9^\circ \text{ und } b_\theta = 14,9^\circ \quad (4.3)$$

verwendet werden.

In Kapitel 3.7.5 wird das Histogramm-Template $\tilde{\mathcal{H}}_T$ als zusätzlicher Parameter verwendet. Es wurde bisher autonom aus den Trainingsdaten bestimmt. Durch die Parameterwahl aus Gleichung 4.3 ergibt sich die Größe der Histogramme $\tilde{\mathcal{H}}_S$ und $\tilde{\mathcal{H}}_T$ zu 3×7 (Winkel $\psi \times \theta \in [-45^\circ, \dots -\frac{b_\psi}{2}, \frac{b_\psi}{2}, \dots 45^\circ] \times [-45^\circ, \dots -\frac{b_\theta}{2}, \frac{b_\theta}{2}, \dots 45^\circ]$). Zur Erstellung der Tabelle 4.16 wurde das Minimum, der Median, das arithmetische Mittel \emptyset und das Maximum aus den jeweiligen Histogrammeinträgen, die für die 15 Lerndatensätze einzeln bestimmt und eingesetzt wurden (siehe Kapitel 3.7.5), des Histogramm-Templates $\tilde{\mathcal{H}}_T$ berechnet. Mit dem jeweiligen Histogramm-Template $\tilde{\mathcal{H}}_T$ (Minimum, Median, \emptyset oder Maximum) und den Parametern aus Gleichung 4.2 und 4.3 wurden die Kostenfunktionen c_{II} und \hat{c}_{II} nach Gleichung 3.112 (Kapitel 3.7.5) berechnet und in Tabelle 4.16 zusammengefasst.

Tabelle 4.16.: Ergebnisübersicht der Detektion mit einheitlichem \mathcal{H}_T

Parametersatz der Segmentierung: $\{T_\psi, T_\theta\}$

Parametersatz der Detektion: $\{b_\psi, b_\theta\}$

	#1		#2		#3		#4		#5	
	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
Min	1,36	2,39	0,73	1,47	1,46	1,40	1,23	1,11	0,92	1,48
Med	1,35	2,39	0,73	1,44	1,44	1,38	1,20	1,09	0,92	1,46
\emptyset	1,34	2,38	0,73	1,44	1,44	1,38	1,20	1,09	0,90	1,45
Max	1,45	2,53	0,84	1,52	1,56	1,55	1,30	1,25	1,06	1,56

	#6		#7		#8		#9		#10	
	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
Min	1,37	1,64	2,11	1,41	2,08	2,15	1,73	1,78	1,23	2,18
Med	1,33	1,62	2,08	1,39	2,05	2,14	1,72	1,72	1,23	2,12
\emptyset	1,33	1,62	2,08	1,39	2,04	2,13	1,72	1,73	1,22	2,12
Max	1,41	1,74	2,22	1,52	2,22	2,24	1,87	1,82	1,34	2,26

	#11		#12		#13		#14		#15	
	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
Min	0,90	1,04	1,02	1,97	0,83	1,23	1,35	1,00	0,79	1,68
Med	0,90	1,02	1,01	1,96	0,81	1,22	1,35	0,98	0,82	1,67
\emptyset	0,90	1,02	1,00	1,96	0,81	1,22	1,35	0,98	0,80	1,66
Max	0,99	1,11	1,18	2,10	0,90	1,30	1,49	1,08	0,96	1,83

Die Einträge der Tabelle 4.16 sind, wie zuvor, farblich gekennzeichnet. Das beste Ergebnis der vier möglichen Histogramm-Templates $\tilde{\mathcal{H}}_T$ (Minimum, Median, \emptyset oder

4. Ergebnisse

Maximum) wurde nach der Kostenfunktion c_{II} , welche auf den Lerndatensätzen bestimmt wurde, in Blau gekennzeichnet, das zweitbeste Ergebnis ist Cyan. Das beste Ergebnis auf den Testdatensätzen nach der Kostenfunktion \hat{c}_{II} ist in Rot und das zweitbeste Ergebnis in Magenta hervorgehoben. In den Spalten der Tabelle 4.16 sind die Ergebnisse für die einzelnen Lern- (c_{II}) bzw. Testdatensätze (\hat{c}_{II}) aufgeführt. Mit 11 bzw. 11 (von 15) besten Ergebnissen nach c_{II} bzw. \hat{c}_{II} ist das arithmetische Mittel \emptyset aus der dritten Zeile der Tabelle 4.16 das Histogramm-Template $\tilde{\mathcal{H}}_T$, welches im Vergleich zum Median, dem Minimum und dem Maximum heraussticht. Der Unterschied zwischen den Ergebnissen ist jedoch so gering, dass der dargestellte Detailgrad der Werte in Tabelle 4.16 im Vergleich zu den Ergebnissen der Tabellen E.29-E.31 aus Anhang E.2.2 erhöht wurde. Zur Verwendung eines einheitlichen Parametersatzes wird das arithmetische Mittel \emptyset der einzelnen Histogramm-Templates $\tilde{\mathcal{H}}_T$ der Lerndatensätze in allen folgenden Auswertungen eingesetzt.

4.5. Ergebnisse der Posenbestimmung und Normalisierung

4.5.1. Parametervergleich

Parametervergleich A

Tabelle 4.17 enthält die nach der Kostenfunktion c_A aus Gleichung 3.116 zur *Parameteroptimierung A* (Kapitel 3.7.6) optimierten Parameter. Die für jeden der 15 Lerndatensätze (1. Spalte) optimierten Werte der Parameter T_N und T_O (siehe Tabelle 3.8 in Kapitel 3.7.6) sind in der jeweiligen Zeile in den Spalten 2 (T_N) und 3 (T_O) der Tabelle 4.17 aufgeführt. Die Spalte 4 der Tabelle 4.17 enthält die mit den jeweiligen Parametern durch c_A über die 199 Frames des jeweiligen Lerndatensatzes bestimmten Kosten. $c_{\emptyset A}$ (Spalte 5) ist der durchschnittliche Wert der Kostenfunktion c_A pro Datensatz: $\frac{c_A}{199}$. \hat{c}_A (Spalte 6 in Tabelle 4.17) ist der Wert, welcher über die Testdatensätze mit den Parametern der Spalten 2 (T_N) und 3 (T_O) nach der Kostenfunktion aus Gleichung 3.116 (Kapitel 3.7.6) bestimmt wurde. Entsprechend wurde $\hat{c}_{\emptyset A}$ durch $\frac{\hat{c}_A}{199}$ bestimmt.

Tabelle 4.17.: Parameter zur *Grobe Posenbestimmung* aus Kapitel 3.5.1

#	T_N	T_O	c_A	$c_{\emptyset A}$	\hat{c}_A	$\hat{c}_{\emptyset A}$
1	95,3 %	98,8 %	4,3 m	2,3 cm	5,7 m	2,9 cm
2	100,0 %	69,8 %	4,7 m	2,4 cm	4,3 m	2,3 cm
3	100,0 %	68,3 %	4,7 m	2,5 cm	4,8 m	2,5 cm
4	97,1 %	69,8 %	4,6 m	2,4 cm	4,5 m	2,3 cm
5	98,4 %	78,0 %	4,3 m	2,2 cm	5,8 m	3,1 cm
6	100,0 %	69,1 %	4,4 m	2,3 cm	4,4 m	2,3 cm
7	97,6 %	67,9 %	4,1 m	2,1 cm	4,7 m	2,4 cm
8	100,0 %	65,2 %	4,7 m	2,4 cm	4,7 m	2,4 cm
9	97,6 %	72,2 %	4,0 m	2,1 cm	4,9 m	2,6 cm
10	96,5 %	71,6 %	3,8 m	1,9 cm	5,2 m	2,7 cm
11	98,3 %	65,7 %	4,4 m	2,3 cm	5,0 m	2,6 cm
12	95,5 %	80,4 %	4,2 m	2,1 cm	4,3 m	2,3 cm
13	93,6 %	80,0 %	3,8 m	2,0 cm	4,7 m	2,4 cm
14	100,0 %	97,4 %	4,6 m	2,4 cm	4,4 m	2,3 cm
15	100,0 %	74,3 %	4,8 m	2,5 cm	5,0 m	2,6 cm

In Tabelle 4.17 fällt auf, dass für einige Lerndatensätze (2, 3, 6, 8, 14 und 15) der Parameter T_N zur Optimierung nach der Kostenfunktion c_A (Gleichung 3.116 aus Kapitel 3.7.6) den maximal zulässigen Wert von 100 % (obere Schranke) annimmt. Die obere Schranke zur *Parameteroptimierung A* (Kapitel 3.7.6) anzupassen wäre jedoch keine Option, da durch $R_d = T_N \max(d_{i,MC})$ mit $T_N = 100\%$ sich $R_d = \max(d_{i,MC})$ ergibt, welches dem höchsten zulässigen Wert für R_d entspricht. Eine entsprechende Beobachtung kann für den Parameter T_O nicht gemacht werden.

4. Ergebnisse

Parametervergleich B

Die nach Gleichung 3.122 der *Parameteroptimierung B* aus Kapitel 3.7.6 optimierten Parameter der Tabelle 3.9 (Kapitel 3.7.6) sind in Tabelle 4.18 aufgeführt. Die für jeden Lerndatensatz (Zeilen der Tabelle) bestimmten Parameter befinden sich in den Spalten 2 – 8 der Tabelle 4.18. Die über die Lerndatensätze nach Gleichung 3.122 (Kapitel 3.7.6) mit den optimierten Parameterwerten bestimmten Kosten c_B sind in der 9 Spalte der Tabelle 4.18 aufgeführt. $c_{\emptyset B} = \frac{c_B}{199}$ in Spalte 10 der Tabelle 4.18 entspricht dem Durchschnittswert der Kostenfunktion c_B umgerechnet auf einen der 199 Datensätze des jeweiligen Lerndatensatzes. Die in den Spalten 11 bzw. 12 (\hat{c}_B bzw. $c_{\hat{\emptyset} B}$) der Tabelle 4.18 aufgeführten Werte wurden über die Testdatensätze bestimmt und nicht, wie bei den entsprechenden Werten der Spalten 9 bzw. 10 (c_B bzw. $c_{\emptyset B}$), über die Lerndatensätze.

Tabelle 4.18.: Parameter zur Posenverfeinerung aus Kapitel 3.5.2

#	$\delta_S, \Delta_S, \Delta_W, \Delta_L, \Delta_A, T_Z$ und δ_{off} angegeben in <i>cm</i>										
	δ_S	Δ_S	Δ_W	Δ_L	Δ_A	T_Z	δ_{off}	c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$
1	1,4	1,8	1,7	24,7	1,4	17,8	5,0	6,9 m	3,6 cm	7,8 m	4,1 cm
2	0,9	1,9	2,3	26,6	0,5	18,5	5,1	6,6 m	3,4 cm	6,9 m	3,6 cm
3	4,1	2,0	1,5	23,7	2,0	19,8	5,2	6,9 m	3,6 cm	7,8 m	4,0 cm
4	0,6	1,9	2,2	24,6	1,4	17,6	5,1	7,3 m	3,8 cm	7,1 m	3,7 cm
5	4,1	1,7	5,2	27,5	1,0	19,7	5,0	6,9 m	3,5 cm	8,8 m	4,6 cm
6	0,9	1,9	2,8	25,5	2,2	17,2	5,0	6,5 m	3,4 cm	7,5 m	3,8 cm
7	0,9	1,9	5,0	29,1	1,9	17,6	5,0	6,7 m	3,5 cm	7,8 m	4,0 cm
8	1,3	1,8	2,7	25,7	0,5	17,8	4,8	6,4 m	3,3 cm	7,2 m	3,7 cm
9	1,2	1,8	14,4	7,6	2,4	18,5	4,8	7,3 m	3,8 cm	8,4 m	4,4 cm
10	1,0	1,5	2,1	21,8	0,8	18,0	4,9	6,6 m	3,4 cm	7,5 m	3,9 cm
11	0,9	1,9	2,6	21,2	4,3	17,1	5,1	7,8 m	4,0 cm	8,2 m	4,3 cm
12	4,1	1,8	1,6	24,5	1,1	19,9	5,0	6,9 m	3,5 cm	7,8 m	4,1 cm
13	2,5	1,9	14,6	7,5	1,9	19,7	5,0	7,1 m	3,7 cm	7,7 m	3,9 cm
14	1,6	1,1	2,0	27,6	0,8	17,1	5,1	6,4 m	3,3 cm	7,0 m	3,5 cm
15	1,0	2,7	14,6	7,8	0,7	19,2	4,9	7,0 m	3,6 cm	7,0 m	3,6 cm

Der Großteil der Parameterwerte in Tabelle 4.18 befinden sich in deutlichem Abstand zu den in der Tabelle 3.10 aus Kapitel 3.7.6 (*Parameter zur Posenverfeinerung B*) angegebenen Schranken. Ausnahmen sind die obere Schranke für Δ_W (Datensätze 9, 13 und 15), die untere Schranke für Δ_A (Datensätze 2 und 8) und die obere Schranke des Parameters T_Z (Datensätze 3, 5, 12, 13 und 15). Ein Zusammenhang zwischen entsprechenden Ausreißern bzw. Parametern nahe der vorgegebenen Schranken und den sich mit diesen ergebenden Kosten c_B bzw. \hat{c}_B (als auch $c_{\emptyset B}$ bzw. $c_{\hat{\emptyset} B}$) ist nicht zu erkennen. Die Werte der Kostenfunktion c_B bzw. \hat{c}_B ($c_{\emptyset B}$ bzw. $c_{\hat{\emptyset} B}$) in Tabelle 4.18 sind im Vergleich zu den Werten der Kostenfunktion c_A bzw. \hat{c}_A ($c_{\emptyset A}$ bzw. $c_{\hat{\emptyset} A}$) in Tabelle 4.17 höher. Dies ist darauf zurückzuführen, dass in der Kostenfunktion c_B bzw. \hat{c}_B ($c_{\emptyset B}$ bzw. $c_{\hat{\emptyset} B}$) neben der Rotation ($\mathbf{R}_{\mathbf{II}+\mathbf{III}}$) die Translation $\mathbf{t}_{\mathbf{II}+\mathbf{III}}$ zur *Parameteroptimierung B* berücksichtigt wird, welches bei der *Parameteroptimierung A* nicht der Fall ist.

4.5.2. Parameterwahl

Das vorherige Kapitel 4.5.1 enthält die Parameter der *Parameteroptimierung A* und der *Parameteroptimierung B* aus Kapitel 3.7.6. Die entsprechenden Parametersätze wurden auf die einzelnen Lerndatensätze angepasst. Um jedoch das Verfahren zur Posenbestimmung aus Kapitel 3.5.1 und 3.5.2 auf unterschiedliche Datensätze (im Idealfall sind diese den bisherigen Daten ähnlich) anwenden zu können, ist es nötig eine Parameterwahl zu treffen, so dass ein fester Parametersatz angewendet werden kann.

Hierzu wurden pro Parameter das Minimum, der Median (siehe Kapitel 3.8.1), das arithmetische Mittel \emptyset (siehe Gleichung 3.115 in Kapitel 3.7.6) und das Maximum, entsprechend der *Parameterwahl* zur Detektion aus Kapitel 4.4.3, berechnet.

Tabelle 4.19 wurde aus den Parametern der Tabelle 4.17 in Kapitel 4.5.1 bestimmt, die aus der *Parameteroptimierung A* (Kapitel 3.7.6) stammen. Die erste Zeile der Tabelle 4.19 enthält das Minimum der Parameterwerte aus Tabelle 4.17 des Parameters T_N (2. Spalte) bzw. T_O (3. Spalte). Die weiteren Zeilen enthalten den Median (2. Zeile), das arithmetische Mittel (3. Zeile) und das Maximum (4. Zeile) der jeweiligen Parameterwerte aus Tabelle 4.17.

Tabelle 4.19.: Parameterübersicht A zur *Grobe Posenbestimmung* aus Kapitel 3.5.1

	T_N	T_O
Min	93,6 %	65,2 %
Med	98,3 %	71,6 %
\emptyset	98,0 %	75,2 %
Max	100,0 %	98,8 %

Entsprechend sind in Tabelle 4.20 das Minimum (1. Zeile), der Median (2. Zeile), das arithmetische Mittel (3. Zeile) und das Maximum (4. Zeile) der durch die *Parameteroptimierung B* bestimmten Werte jedes Parameters aus Tabelle 4.18 (Kapitel 4.5.1) zusammengefasst.

Tabelle 4.20.: Parameterübersicht B zur Posenbestimmung aus Kapitel 3.5.2

	δ_S	Δ_S	Δ_W	Δ_L	Δ_A	T_Z	δ_{off}
Min	0,6 cm	1,1 cm	1,5 cm	7,5 cm	0,5 cm	17,1 cm	4,8 cm
Med	1,2 cm	1,9 cm	2,6 cm	24,6 cm	1,4 cm	18,0 cm	5,0 cm
\emptyset	1,8 cm	1,8 cm	5,0 cm	21,7 cm	1,5 cm	18,4 cm	5,0 cm
Max	4,1 cm	2,7 cm	14,6 cm	29,1 cm	4,3 cm	19,9 cm	5,2 cm

Zur Parameterwahl wurden für alle Kombinationen der Parametersätze aus Tabelle 4.19 (*Parameteroptimierung A*) und Tabelle 4.20 (*Parameteroptimierung B*) die Kostenfunktion c_B bzw. \hat{c}_B nach Gleichung 3.122 (Kapitel 3.7.6) mit den Lern- bzw. Testdatensätzen berechnet. Die direkten Ergebnisse sind im Anhang F in den Tabellen F.1-F.5 zu finden. In den Tabellen F.1-F.5 wurden pro Lern- bzw. Testdatensatz die besten bzw. zweitbesten Ergebnisse markiert. Die Anzahl der besten bzw.

4. Ergebnisse

zweitbesten Ergebnisse aus den insgesamt 15 Lerndatensätze ist in Tabelle 4.21 pro Kombination der Parametersätze (Spalten 1 und 2 der Tabelle 4.21) in der Spalte 3 (N_{c_B}) in Blau bzw. Cyan gekennzeichnet. Entsprechend ist die Anzahl $N_{c_{\hat{B}}}$ (Spalte 4 der Tabelle 4.21) der besten bzw. zweitbesten Ergebnisse, welche sich mit der jeweiligen Parametersatzkombination (Spalte 1 und 2) aus den 15 Testdatensätzen ergeben, in Rot bzw. Magenta eingetragen.

Tabelle 4.21.: Ergebnisübersicht der Posenbestimmung der Kapitel 3.5.1 und 3.5.2

A	B	N_{c_B}	$N_{c_{\hat{B}}}$
Min	Min	-/-	-/-
Min	Med	1/-	-/-
Min	\emptyset	-/-	-/-
Min	Max	-/-	-/-
Med	Min	-/-	-/-
Med	Med	-/2	-/1
Med	\emptyset	-/1	-/-
Med	Max	-/-	-/-
\emptyset	Min	-/-	-/-
\emptyset	Med	1/2	-/-
\emptyset	\emptyset	-/-	-/-
\emptyset	Max	-/-	-/-
Max	Min	1/-	-/-
Max	Med	11/2	10/5
Max	\emptyset	1/8	5/9
Max	Max	-/-	-/-

In der Übersicht aus Tabelle 4.21 stechen zwei Parameterkombinationen aus den Zeilen 14 (A: Maximum & B: Median) und 15 (A: Maximum & B: arithmetisches Mittel \emptyset) hervor. Sowohl für die Lern- als auch für die Testdatensätze hat das Maximum aus Tabelle 4.19 kombiniert mit dem Median aus Tabelle 4.20 als kompletter Parametersatz (Zeile 14 in Tabelle 4.21) die meisten besten Ergebnisse (11 bzw. 10). Zudem zählen insgesamt 13 bzw. 15 der Lern- bzw. Testdatensätze mit dem Parametersatz (A: Maximum & B: Median) zu den besten zwei Ergebnissen.

Zur weiteren Verwendung werden die Parameter zur Posendetektion nach diesem Parametersatz (A: Maximum & B: Median) festgelegt: $T_N = 100\%$, $T_O = 98.8\%$, $\delta_S = 1.2\text{ cm}$, $\Delta_S = 1.9\text{ cm}$, $\Delta_W = 2.6\text{ cm}$, $\Delta_L = 24.6\text{ cm}$, $\Delta_A = 1.4\text{ cm}$, $T_Z = 18.0\text{ cm}$ und $\delta_{off} = 5.0\text{ cm}$.

4.5.3. Parameter zur Normalisierung der 2D Daten

Die Kameramatrix \mathbf{K}_V wurde nach dem in Kapitel 3.7.6 beschriebenen Vorgehen bestimmt. Der Pixelabstand der Landmarken Δ_{px} wurde zuvor auf $60\,px$ festgelegt, um eine Bildgröße von 80×175 bei den normalisierten Bildern zu erzeugen. Der intrinsische Parameter α_v (siehe Kapitel 3.5.3) der Matrix \mathbf{K}_V wurde mit $\alpha_v = 336,94$ festgelegt. $x_v = 40$ richtet die Pferdeköpfe in horizontaler Richtung zentriert auf den normalisierten Bildern aus und $y_v = 35$ lässt in vertikaler Richtung nach oben genügend Platz für die Bereiche des Kopfes, welche sich oberhalb der Landmarken befinden.

Die Matrix

$$\mathbf{K}_V = \begin{bmatrix} \alpha_v & 0 & x_v \\ 0 & \alpha_v & y_v \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 336,94 & 0 & 40 \\ 0 & 336,94 & 35 \\ 0 & 0 & 1 \end{bmatrix} \quad (4.4)$$

wird nach Kapitel 3.5.3, neben der durch die Pose bestimmten Rotation $\mathbf{R}_{\mathbf{I}+\mathbf{II}+\mathbf{III}}$ und Translation $\mathbf{t}_{\mathbf{I}+\mathbf{II}+\mathbf{III}}$, dazu genutzt aus den ursprünglichen Grauwertbildern der Kameras die normalisierten, virtuellen Kamerabilder zu generieren.

4.6. Ergebnisse der Identifizierung individueller Pferde

4.6.1. Daten zur Identifikation

Tabelle 3.4 aus Kapitel 3.7.2 führt die Anzahl der Besuche mit den jeweiligen Zeiträumen und entsprechender Beleuchtungsart auf, die mit dem Bilderfassungssystem aus Kapitel 3.3 aufgezeichnet wurden. Die während dieser Besuche aufgezeichneten Frames (Bild der Tiefenkamera, Farbkamera und der zwei Industriekameras) bilden die Datenbasis zu den in dieser Arbeit vorgestellten Verarbeitungsschritten. In den Disparitätsbildern der Tiefenkamera aller Besuche aus Tabelle 3.4 wurden zunächst nach der Beschreibung in Kapitel 3.7.7 alle Pferdeköpfe aufgefunden und extrahiert. Dabei kamen die Verfahren aus Kapitel 3.4.3 und 3.4.4 mit den Parametern aus Kapitel 4.4.3 zum Einsatz. Anschließend wurde eine *Symmetrieprüfung* nach Kapitel 3.7.7 durchgeführt.

Die Anzahl der nach der *Symmetrieprüfung* in den Daten vorhandenen Frames, bei denen nach der Klassifikation mit $T_{ROC} = 0,35$ (siehe Kapitel 3.7.7) ein Pferdekopf im Bild erwartet wird, ist in Tabelle 4.22 pro Pferd und Aufnahmezeitraum zusammengefasst. In den ersten beiden Spalten der Tabelle 4.22 sind in jeder Zeile die jeweiligen Aufnahmezeiträume (Zeitraum von, bis) aufgeführt. Die Spalten 3 und 4 der Tabelle 4.22 geben an, welche der beiden Leuchtstoffröhren während der Aufnahmen in Betrieb waren. Alle folgenden Spalten 5 – 18 geben in der Tabelle 4.22 je Pferd die Anzahl der extrahierten Frames wieder. Der Wert innerhalb der Klammern, welcher unterhalb jeder Frameanzahl angegeben ist, gibt den prozentualen Anteil der extrahierten zu den aufgezeichneten Frames an. Der in den Klammern angegebene Anteil liegt in der Tabelle 4.22 zwischen 5,1 % und 28,1 % und hängt stark vom Verhalten der Tiere ab, wodurch dieser nicht repräsentativ ist.

Die Frames der Tabelle 4.22 wurden anschließend manuell nach der Beschreibung in *Gruppieren der Daten* aus Kapitel 3.7.7 in die Gruppen „Gut“, „Schlecht“, „Sehr Schlecht“, „Sonstiges“ und „Kein Kopf“ eingeteilt. Die Anzahl der Frames aus jeder Gruppe sind in den Tabellen 4.23 und 4.24 untereinander aufgeführt. Die jeweiligen Gruppen werden dabei in den Tabellen 4.23/4.24 farblich unterschieden: die Gruppe „Gut“ ist in Grün dargestellt, „Schlecht“ in Schwarz, „Sehr Schlecht“ in Rot, „Sonstiges“ in Blau und die Gruppe „Kein Kopf“ ist in Dunkelgrau dargestellt.

Die Zeilen aus denen die Frames der Gruppe „Gut“ stammen, welche zur Auswertung in Kapitel 4.6.2 verwendet werden, sind in Grau hinterlegt. Die Datensätze wurden entsprechend der Beschreibung aus *Daten zur Identifikation* in Kapitel 3.7.7 ausgewählt; die entsprechenden Pferdeköpfe sind durch die verwendeten Leuchtstoffröhren von beiden Seiten beleuchtet.

4.6. Ergebnisse der Identifizierung individueller Pferde

Tabelle 4.22.: Übersicht der Anzahl detektierter Frames mit Pferdekopf (% der gesamten Frames)

Zeitraum		Beleuchtung		Anna	Arielle	Baldo	Bara	Blüte	Blütenzauber	Della	Diadem	Dumblodore	Linus	Romanze	Rosi	Royal Bess	Serenade
von	bis	links	rechts														
10.11. 2011	10.11. 2011	x	x	-	618 (17,8)	-	80 (10,0)	-	-	-	-	-	-	-	-	-	-
13.04. 2012	13.04. 2012	x	x	42 (15,1)	67 (10,8)	-	107 (16,3)	75 (13,1)	99 (26,2)	-	19 (14,0)	-	-	74 (12,0)	36 (7,1)	-	-
24.07. 2012	11.08. 2012	x	x	-	-	172 (10,1)	-	-	-	-	-	-	-	-	1.089 (10,2)	942 (9,9)	946 (8,7)
10.12. 2012	02.01. 2013	x	x	2.531 (19,5)	1.816 (13,0)	-	1.157 (14,3)	3.242 (21,8)	1.883 (15,8)	2.012 (14,7)	-	1.305 (16,9)	-	2.221 (14,9)	1.041 (7,8)	-	-
04.01. 2013	10.01. 2013	x		2.486 (20,2)	-	-	1.242 (12,6)	1.467 (18,7)	1.207 (13,7)	1.066 (11,6)	-	1.770 (17,4)	-	1.232 (13,5)	308 (5,1)	-	-
11.01. 2013	27.01. 2013		x	2.072 (28,1)	2.659 (23,8)	-	1.760 (17,4)	3.077 (28,1)	1.479 (18,2)	1.908 (16,4)	-	1.947 (23,2)	-	1.389 (16,4)	1.183 (11,6)	-	-
28.01. 2013	28.02. 2013	x	x	4.508 (19,2)	3.320 (15,1)	-	2.844 (14,8)	3.627 (18,0)	2.444 (14,3)	4.178 (12,9)	-	3.753 (17,3)	-	2.674 (13,1)	1.861 (9,3)	-	-
23.04. 2013	26.04. 2013	x		-	407 (14,2)	-	379 (14,4)	-	170 (8,8)	203 (11,9)	-	205 (23,5)	651 (19,9)	131 (7,2)	135 (5,4)	-	-
26.04. 2013	30.04. 2013		x	-	291 (15,0)	-	262 (15,8)	-	211 (14,0)	93 (6,3)	-	407 (19,0)	449 (22,9)	177 (12,4)	45 (6,9)	-	-
30.04. 2013	06.05. 2013	x	x	-	1.216 (12,3)	-	-	-	-	716 (9,9)	-	974 (15,4)	302 (12,8)	671 (11,7)	339 (5,1)	-	-
06.05. 2013	07.05. 2013		x	-	421 (15,9)	-	-	-	-	266 (11,3)	-	695 (19,5)	506 (21,9)	159 (12,4)	146 (5,6)	-	-

4. Ergebnisse

Tabelle 4.23.: Übersicht der Anzahl gruppierter Frames (1)

Zeitraum		Beleuchtung links	rechts	Anna	Arielle	Baldo	Bara	Blüte	Blütenzauber	Delia
von	bis									
10.11. 2011	10.11. 2011	x	x	- -	94 74 450	- -	7 34 39	- -	- -	- -
13.04. 2012	13.04. 2012	x	x	17 12 13	34 18 14	- -	37 41 28	23 18 31	43 23 26	- -
25.07. 2012	31.07. 2012	x	x	- -	- -	- 5 101	- -	- -	- -	- -
10.12. 2012	02.01. 2013	x	x	777 569 965	852 336 412	- -	515 311 270	913 826 1.194	611 546 656	499 835 485
04.01. 2013	10.01. 2013	x		4 216	9 207	-	61	56 253	6 64	- 193
11.01. 2013	27.01. 2013		x	887 638 941	- -	- -	570 328 309	438 497 472	417 308 449	246 508 291
28.01. 2013	28.02. 2013	x	x	6 14	-	-	- 35	- 60	- 33	- 21
23.04. 2013	26.04. 2013	x		451 375 743	920 450 394	- -	386 287 374	503 874 924	248 223 566	274 696 392
26.04. 2013	30.04. 2013		x	1 502	5 890	-	1 712	- 776	- 442	4 542
30.04. 2013	06.05. 2013	x	x	1.553 1.072 1.352	1.027 534 948	- -	646 413 711	675 1.076 1.158	479 392 840	556 1.540 729
06.05. 2013	07.05. 2013		x	51 480	239 572	-	259 815	28 690	247 486	164 1.189
23.04. 2013	26.04. 2013	x		- -	157 108 94	- -	174 66 107	- -	48 25 53	54 92 53
26.04. 2013	30.04. 2013		x	- -	83 67 99	- -	109 60 57	- -	74 59 65	16 39 32
30.04. 2013	06.05. 2013	x	x	- -	350 270 421	- -	- -	- -	- -	54 221 304
06.05. 2013	07.05. 2013		x	- -	247 67 106	- -	- -	- -	- -	42 108 112
					- 1					1 3

4.6. Ergebnisse der Identifizierung individueller Pferde

Tabelle 4.24.: Übersicht der Anzahl gruppierter Frames (2)

Zeitraum		Beleuchtung links	rechts	Diadem	Dumbledore	Linus	Romanze	Rosi	Royal Bess	Serenade
von	bis									
13.04. 2012	13.04. 2012	x	x	11 6 2 -	- - -	- - -	30 26 18 -	5 21 10 -	- - -	- - -
24.07. 2012	11.08. 2012	x	x	- - -	- - -	- - -	- - -	182 430 313 33 131	34 244 445 10 209	184 378 240 - 144
10.12. 2012	02.01. 2013	x	x	- - -	264 249 610 1 181	- - -	716 475 782 1 247	269 432 202 - 138	- - -	- - -
04.01. 2013	10.01. 2013	x		- - -	413 367 932 - 58	- - -	397 299 503 1 32	56 118 116 - 18	- - -	- - -
11.01. 2013	27.01. 2013		x	- - -	231 303 720 - 693	- - -	320 305 393 - 371	65 236 151 - 731	- - -	- - -
28.01. 2013	28.02. 2013	x	x	- - -	1.063 843 1.155 290 402	- - -	633 592 858 163 428	143 456 352 42 868	- - -	- - -
23.04. 2013	26.04. 2013	x		- - -	63 57 66 - 19	86 260 179 1 125	34 46 48 - 3	22 31 49 - 33	- - -	- - -
26.04. 2013	30.04. 2013		x	- - -	158 93 130 - 26	104 160 135 - 50	50 59 52 - 16	6 25 11 - 3	- - -	- - -
30.04. 2013	06.05. 2013	x	x	- - -	383 326 246 - 19	20 48 152 76 6	106 191 192 51 131	21 132 135 38 13	- - -	- - -
06.05. 2013	07.05. 2013		x	- - -	302 213 172 - 8	121 221 161 - 3	50 65 44 - -	33 55 56 - 2	- - -	- - -

4. Ergebnisse

Abbildung 4.13 zeigt die Anzahl der Frames in den einzelnen Gruppen, ohne die Aufschlüsselung der einzelnen Tiere, übereinander dar. Dabei werden verschiedene Schwellwerte verwendet, um die Auswahl der Frames in der Gruppe „Gut“ (bzw. „Gut“ und „Schlecht“) für die Praxis darzustellen und die Verteilung der Daten in Abhängigkeit bestimmter Schwellen aufzuzeigen. Die Gruppen sind farblich entsprechend der Einträge aus den Tabellen 4.23/4.24 gekennzeichnet.

In Abbildung 4.13a ist der zur *Detektion* verwendete Schwellwert T_ρ auf dem Ähnlichkeitsfaktor ρ aus Gleichung 3.55 (Kapitel 3.4.4) auf der Abszisse aufgetragen. Umso kleiner der Schwellwert T_ρ für ρ ($\rho < T_\rho \equiv T_{ROC}$, siehe Kapitel 3.7.7 bzw. Tabellen 4.10 und 4.11 aus Kapitel 4.4.2) gewählt wird, umso geringer ist die Anzahl der Frames in den einzelnen Gruppen, welche unterhalb des Schwellwerts T_ρ liegen. Auf der rechten Seite der Abbildung 4.13a sind die gesamten 89.662 manuel gruppierten Frames aus den Tabellen 4.23 und 4.24, bei der Verwendung des Schwellwerts $T_\rho = 0,35$ ($\rho \leq T_\rho$), zu erkennen. Nach links hin wird der auf den Ähnlichkeitsfaktor ρ angewendete Schwellwert kleiner und mit ihm die Anzahl der Frames in den jeweiligen Gruppen, welche nicht durch den Schwellwert ausgeschlossen wurden und auf der Ordinate der Abbildung 4.13a aufgetragen sind. Die magenta farbene Linie markiert den Schwellwert $T_\rho = 0,25$, der für die Auswahl in Abbildung 4.14 auf ρ angewendet wurde.

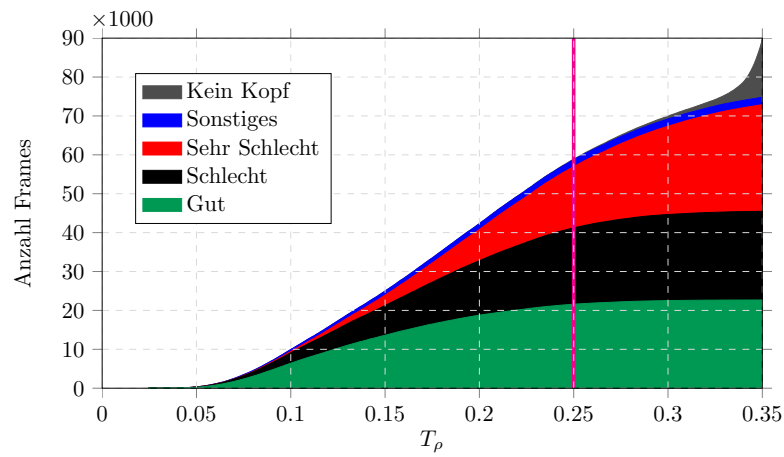
Abbildung 4.13b zeigt die Anzahl der nach den 5 Gruppen sortierten Frames, nach Anwendung des Schwellwerts $T_{V_{px}}$ auf die Anzahl der Pixel (maskierte Pixel nach *Daten zur Identifikation* in Kapitel 3.7.7) im virtuellen Kamerabild (siehe Kapitel 3.5.3) mit einer Auflösung von 80×175 . Der Schwellwert $T_{V_{px}}$, welcher durch die Linie in Magenta markiert ist, liegt bei 6.500 Pixeln und wurde für die Auswahl in Abbildung 4.14 verwendet. Die 6.500 Pixel entsprechen auf einer 1 m entfernten Ebene (entspricht der Entfernung der normalisierten Pferdeköpfe) einer Fläche von $551,7 \text{ cm}^2$ und decken 46,4 % des virtuellen Bilds ab. Sollte die Auflösung des virtuellen Grauwertbilds oder der Abstand des Kopfes zum Rand der Bilder verändert werden, so ist der Schwellwert $T_{V_{px}}$ entsprechend anzupassen. Die Anzahl der Pixel im virtuellen Kamerabild dient vorrangig dazu die Frames der Gruppe „Sehr Schlecht“ von den restlichen Frames zu trennen. Der Schwellwert $T_{V_{px}}$ ist dazu im Fall der Abbildung 4.13b umgekehrt (Pixel der virtuellen Kamera $> T_{V_{px}}$) dem Schwellwert T_ρ (aus Abbildung 4.13a) anzuwenden, um die Frames der Gruppe „Gut“ (bzw. „Gut“ und „Schlecht“) herausfiltern zu können.

Pferdeköpfe, die nicht komplett im Tiefenbild liegen, beeinträchtigen die autonome Posenbestimmung der Verfahren aus Kapitel 3.5. Ist die Pose nicht korrekt, so wird auch das virtuelle Kamerabild, in dem der Pferdekopf in die Kamera schauen sollte, keine guten Bilder für eine Identifikation des Tieres erzeugen. Entsprechende Frames wurden in Abbildung 4.13c über die Pixelanzahl in der Maske $\mathbf{M} \oplus \mathbf{S}$ (siehe Abbildung 3.34a aus Kapitel 3.5.2) bestimmt, die in einem Abstand von zwei Pixeln zum Rand des Disparitätenbilds liegen. Werden in dem Rand von zwei Pixeln

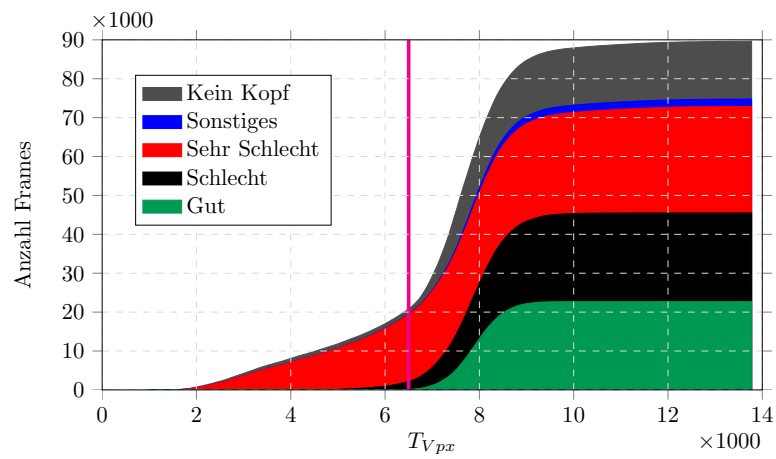
Teile des Kopfes detektiert, so ist es wahrscheinlich, dass der Kopf nicht komplett im Tiefenbild abgebildet ist. Entsprechend ist die magenta farbene Linie auf dem Schwellwert $T_{Bpx} = 0$ gelegt, so dass Frames ausgeschlossen werden, die auch nur einen Pixel in dem genannten Randbereich haben. In der Abbildung 4.13c ist gut zu erkennen, dass dieses Auswahlkriterium einen äußerst geringen Einfluss auf die Anzahl der Frames in der Gruppe „Gut“ hat.

Abbildung 4.14 enthält die Anzahl der Frames pro Gruppe als Balkendiagramm dargestellt. Es zeigt, wie sich die Frames mit nur der Beschränkung der Frames durch den Ähnlichkeitsfaktor ρ ($\rho \leq T_\rho = 0,35$) „Vorher“ und einer Kombination der in Abbildung 4.13 dargestellten Ausschlussbedingungen „Nachher“ zusammensetzten. Die Ausschlussbedingungen sind beispielhaft gewählt und können den entsprechenden Bedingungen angepasst werden. Die Anzahl der Frames, dargestellt durch die Balken nach der Ausschlussprüfung „Nachher“, wurde durch die in magenta hervorgehobenen Schwellwerte aus Abbildung 4.13 bestimmt. Der Ähnlichkeitsfaktor ρ liegt unterhalb von $\rho < T_\rho = 0,25$, die Anzahl der Pixel im virtuellen Kamerabild mit einer Auflösung 80×175 liegt oberhalb von $T_{Vpx} = 6.500$ und die Anzahl der Pixel auf dem Rand des Disparitätenbilds liegt bei $T_{Bpx} = 0$ (Anzahl Pixel ≤ 0).

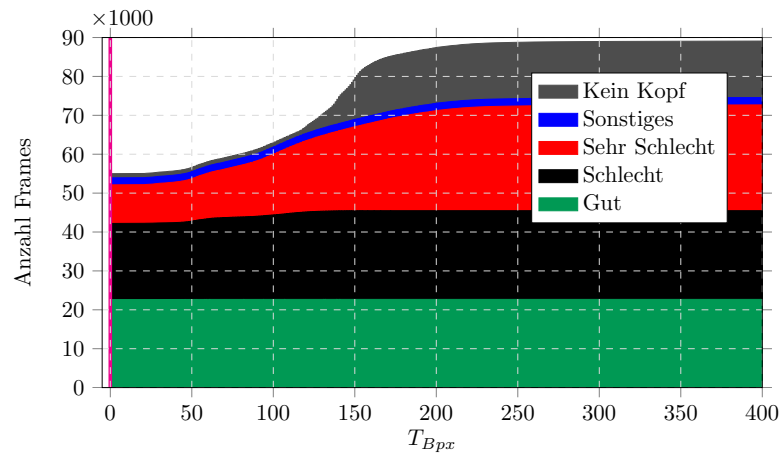
4. Ergebnisse



(a) Ähnlichkeitsfaktor ρ der Detektion



(b) Pixel im virtuellen Kamerabild



(c) Anzahl Randpixel

Abbildung 4.13.: Frames pro Gruppe mit Ausschlussbedingung

4.6. Ergebnisse der Identifizierung individueller Pferde

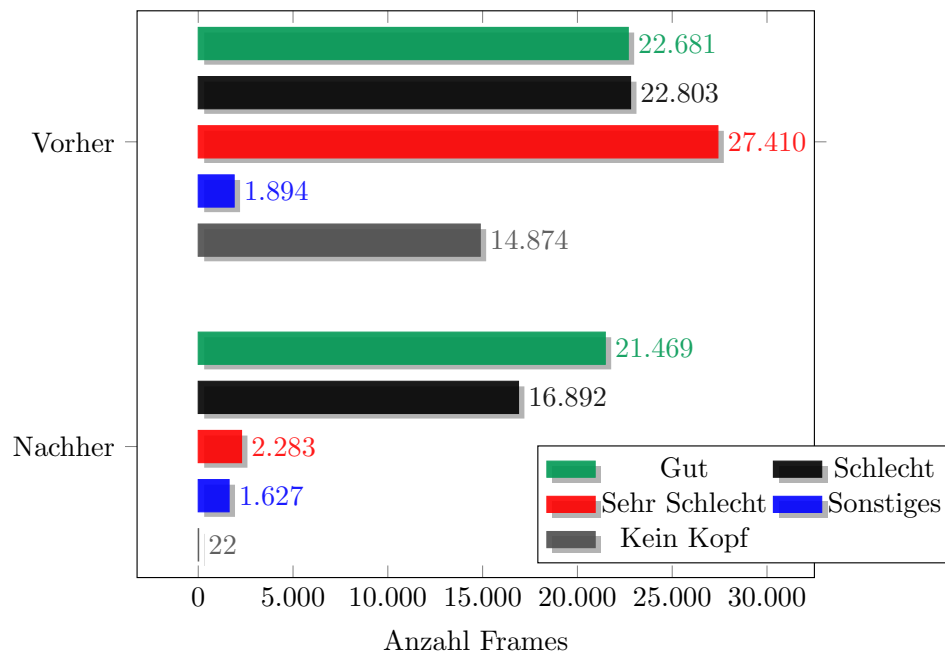


Abbildung 4.14.: Frames pro Gruppe vor und nach Ausschlussprüfung

Mit der vorgestellten Ausschlussprüfung, die in Abbildung 4.14 dargestellt ist, lassen sich die Frames der Gruppen „Gut“ und „Schlecht“ auf den Daten der Tabellen 4.23 und 4.24 sehr gut von den restlichen Gruppen trennen.

4.6.2. Identifikationsergebnisse

In Kapitel 3.7.7 wird beschrieben wie die Ergebnisse der Identifikation von einzelnen Individuen bei Pferden erzeugt wurden. Dieses Kapitel stellt dabei die unterschiedlichen Variationen der Auswertungen auf den mit der zweiten Industriekamera (siehe Aufbau aus Kapitel 3.3) aufgenommenen Daten zusammen. Entsprechende Auswertungen zu der Farb- und ersten Industriekamera (siehe Aufbau aus Kapitel 3.3) werden aufgrund des Umfangs nicht weiter behandelt. Die in diesem Kapitel vorgestellte Unterscheidung der Pferde anhand ihrer Gesichter basiert, entsprechend der Beschreibung aus Kapitel 3.7.7, auf den Daten einer Gruppe von 9 Pferden. Die Pferde in dieser Gruppe sind nochmals in Abbildung 4.15 gezeigt.



Abbildung 4.15.: Die 9 Pferde der Test- und Validierungsgruppe

Abbildung 4.16 zeigt die Treffergenauigkeit (ACC) nach Gleichung 3.136 aus Kapitel 3.8.3 der Multi-Klassen Entscheidung der Methode aus Kapitel 3.6.2, welche gegen den verwendeten Anteil der Lerndaten (prozentual von jeweils 200 Bildern pro Tier) aufgetragen ist. Der Anteil der verwendeten Lerndaten gibt damit die Anzahl der Bilder an, die zum Anlernen und Evaluieren verwendet wurde (vgl. Kapitel 3.7.7). Zudem sind die eingesetzten Bilddaten für jede der zu den einzelnen Variationen der verwendeten Bildgröße bzw. der verwendeten Eigenwerte (EW) eingezeichneten Kurven aus Abbildung 4.16 in vertikaler Richtung (gleicher Anteil von Lerndaten) identisch.

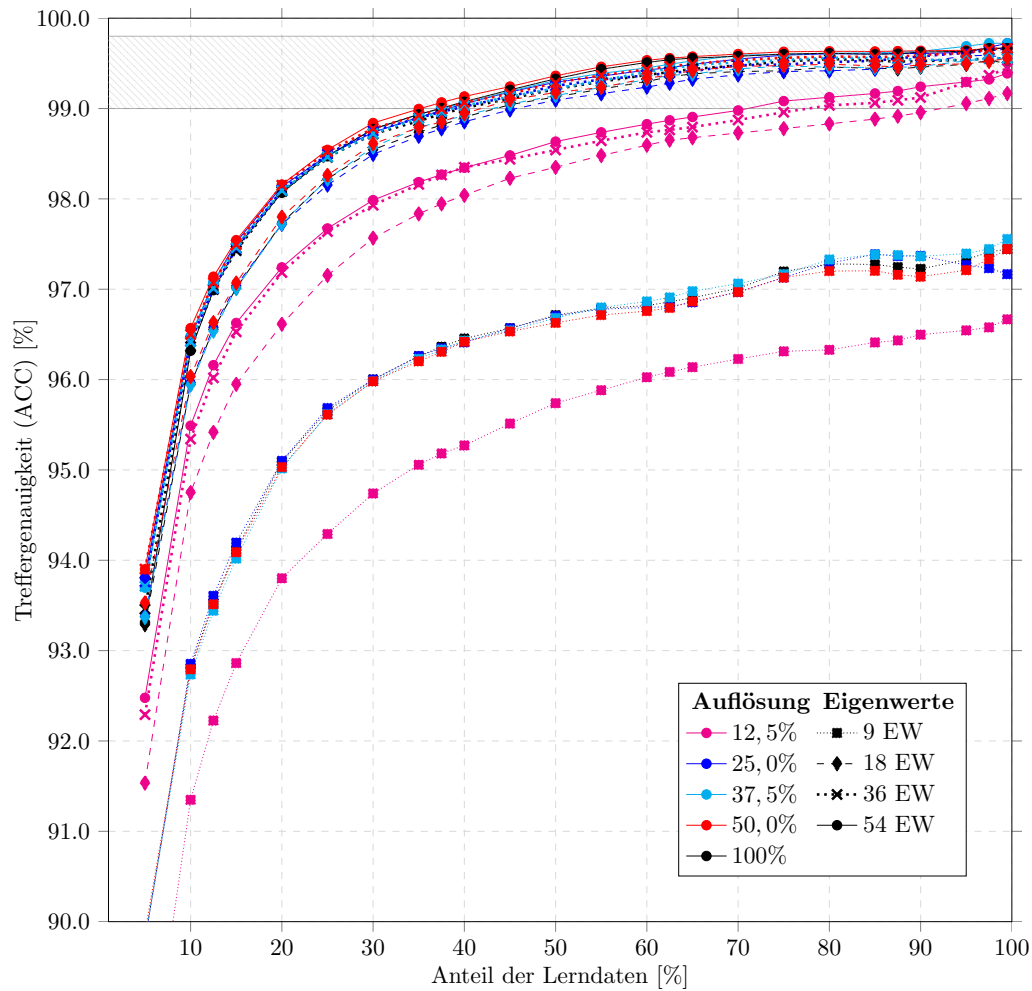


Abbildung 4.16.: Identifikationsergebnisse der zweiten Industriekamera

Die Variation der Bildgröße (Auflösung variiert von 10×22 bis 80×175) ist in den Graphen der Abbildung 4.16 durch unterschiedliche Farben gekennzeichnet. Die eingesetzte Auflösung ist prozentual ausgehend von einer Auflösung von 80×175 (100%) in der Legende angegeben. Die Variation der eingesetzten Eigenwerte (9, 18, 36 bzw. 54 EW) ist durch die Linienart (fein gepunktet 9 EW, gestrichelt 18 EW, dicker gepunktete 36 EW bzw. durchgehend 54 EW) der Kurven aus Abbildung 4.16 zu unterscheiden, als auch durch die verwendete Markierung in jedem Punkt der Kurven.

Die unteren fünf Kurven geben dabei die Identifikationsergebnisse mit 9 Eigenwerten an, wovon die unterste Kurve das Ergebnis mit der niedrigsten Auflösung von 10×22 (12%) in Magenta darstellt. Ausgehend von diesen fünf fein gepunkteten Kur-

4. Ergebnisse

ven (9 EW), sind die nächsten drei darüber liegenden Kurven allesamt in Magenta gezeichnet, so dass der Einsatz von 9 Eigenwerten schlechtere Ergebnisse liefert, als die Identifikation mit der hier eingesetzten geringsten Auflösung von 10×22 (12%), wenn mehr als 9 Eigenwerte eingesetzt werden. Werden die unteren acht Kurven, bei denen eine Auflösung von 10×22 (12%) bzw. 9 Eigenwerte eingesetzt wurden, ausgeblendet, so lassen sich die nächsten vier Kurven den Ergebnissen mit 18 Eigenwerten und den Auflösungen über 10×22 (12%) zuordnen. Damit hebt sich der Einsatz von 9 Eigenwerten jeweils stark von den restlichen Kurven ab, wobei sich der Einsatz von 18 Eigenwerten nur geringfügiger von den verbleibenden Kurven abhebt. Entsprechendes gilt für den Einsatz der niedrigsten Auflösung von 10×22 (12%), wobei hier die Ergebnisse deutlich näher an den verbleibenden Ergebnissen liegen. Die verbleibenden Kurven bei denen mehr als 18 Eigenwerte verwendet wurden, lassen sich in der linearen Darstellung aus Abbildung 4.16 nur schlecht auseinander halten.

Der in Abbildung 4.16 in Grau schraffierte Bereich oberhalb einer Treffergenauigkeit von 99% und unterhalb von 99,8% ist in Abbildung 4.17 nochmals vergrößert dargestellt. Die Kennzeichnung der Kurven ist dabei identisch zu der aus Abbildung 4.16. Mit wenigen Ausnahmen lassen sich die Kurven aus Abbildung 4.17 nach den Eigenwerten sortiert so interpretieren, dass der Einsatz von mehr Eigenwerten auch bessere Ergebnisse liefert, wobei die Auflösung (mit Ausnahme der kleinsten Auflösung 10×22) einen geringeren Einfluss zeigt.

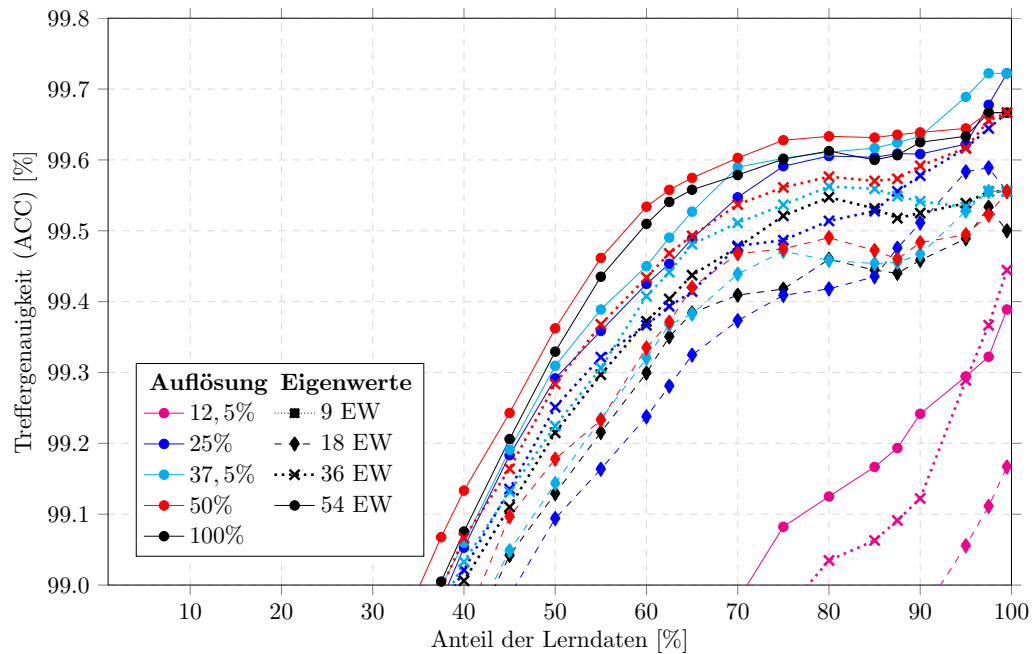


Abbildung 4.17.: Identifikationsergebnisse der zweiten Industriekamera (vergrößert)

Die Kurven der höchsten Auflösung (100%) beim Einsatz von 36 bzw. 54 Eigenwerten liegen in Abbildung 4.17 (mit Lernanteil größer 40%) sogar unterhalb der Kurven mit einer Auflösung von 37,5% (30×66) bzw. 50% (40×88) bei gleicher Anzahl von eingesetzten Eigenwerten.

In den folgenden Beobachtungen sind die Kurven mit der geringsten Auflösung 10×22 (12%) und solche, die mit 9 Eigenwerten erzeugt wurden, ausgeschlossen. Beim Einsatz von mehr als 35% der Lerndaten, ließ sich für die Kurven mit jeweils gleichbleibender Auflösung (25%, 37,5%, 50% bzw. 100%), zwischen den Kurven, in denen pro Auflösung eine unterschiedliche Anzahl (18, 36 oder 54) von Eigenwerten verwendet wurde, eine absolute Differenz der Treffergenauigkeit von unter 0,23% feststellen. Für den Einsatz eines beliebigen Anteils von Lerndaten ergibt sich für diese Kurven (gleichbleibende Auflösung, unterschiedliche Anzahl Eigenwerte) eine maximale absolute Abweichung von 0,54% in der Treffergenauigkeit. Bleibt hingegen die Anzahl (18, 36 oder 54) der verwendeten Eigenwerten konstant, so liegt die absolute Abweichung der Treffergenauigkeit für die jeweiligen Kurven, bei unterschiedlicher Auflösung (25%, 37,5%, 50% bzw. 100%), unter 0,12%, solange mehr als 15% der Lerndaten genutzt werden. Wird der Lernanteil beliebig gewählt, so liegt die absolute Differenz für diese Kurven (gleichbleibende Anzahl an Eigenwerten, unterschiedliche Auflösung) unter 0,6% der Treffergenauigkeit.

Die folgenden Auswertungen wurden auf eine Auflösung von 30×66 (37,5%) und den Einsatz von 36 Eigenwerten beschränkt, da die Ergebnisse nach Abbildung 4.16 bzw. 4.17 im oberen Mittelfeld liegen und die nähere Auswertung aller Kurven den Rahmen dieser Dissertation übersteigt. Abbildung 4.18 zeigt die CMC-Kurven (*cumulative match curves*) (siehe Kapitel 3.8.3) mit der genannten Auflösung (30×66) und den 36 Eigenwerten für unterschiedliche Lernanteile. Der Anteil der Lerndaten ist in Abbildung 4.18 durch die jeweilige Farbe und Linienart der Kurve (siehe Legende) gekennzeichnet. Auf der Abzisse findet sich der Rang, welcher, entsprechend der Beschreibung aus Kapitel 3.8.3, gegen die Erkennungsrate auf der Ordinate aufgetragen ist. In Abbildung 4.18 lassen sich daher die Erkennungsraten nach dem Rang ablesen, wobei die Erkennungsraten des ersten Rang (ROR - *rank one recognition rate*) bereits in den Abbildungen 4.16 und 4.17 angegeben sind.

In Abbildung 4.18 ist ein deutlicher Anstieg vom ersten auf den zweiten Rang für alle eingezeichneten Kurven erkennbar, wobei dieser größer ist je kleiner der verwendete Anteil der Lerndaten ist. Aufgrund der Übersicht sind nicht alle in den Abbildungen 4.16 bzw. 4.17 eingetragenen Lernanteile in Abbildung 4.18 als Kurven eingezeichnet. Die gewählten Lernanteile sind in der Legende der Abbildung zu erkennen, wobei die Kurve mit einem Lernanteil von 75% (150 Bilder) kaum von der mit einem Lernanteil von 99,5% (199 Bilder) zu unterscheiden ist. Die Kurve mit einem Lernanteil von 50% (100 Bildern) ist ebenfalls recht nahe an den beiden oberen Kurven (75 und 99,5% Lernanteil) und ab dem dritten Rang kaum noch von diesen zu unterscheiden. Es folgen die Kurven mit geringeren Lernanteilen, wobei

4. Ergebnisse

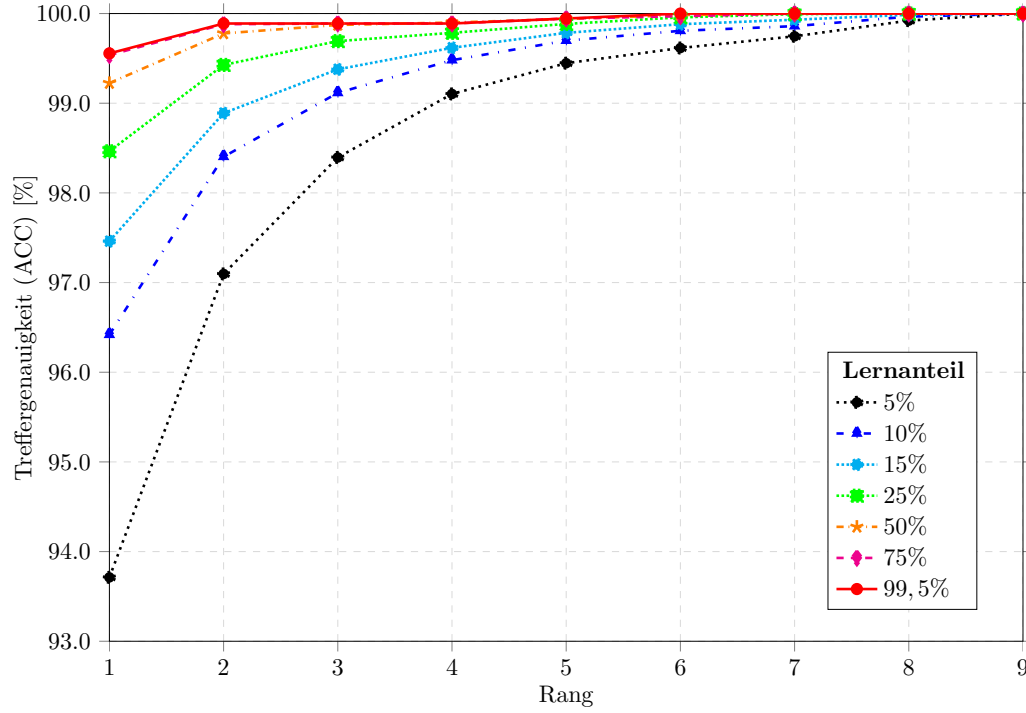


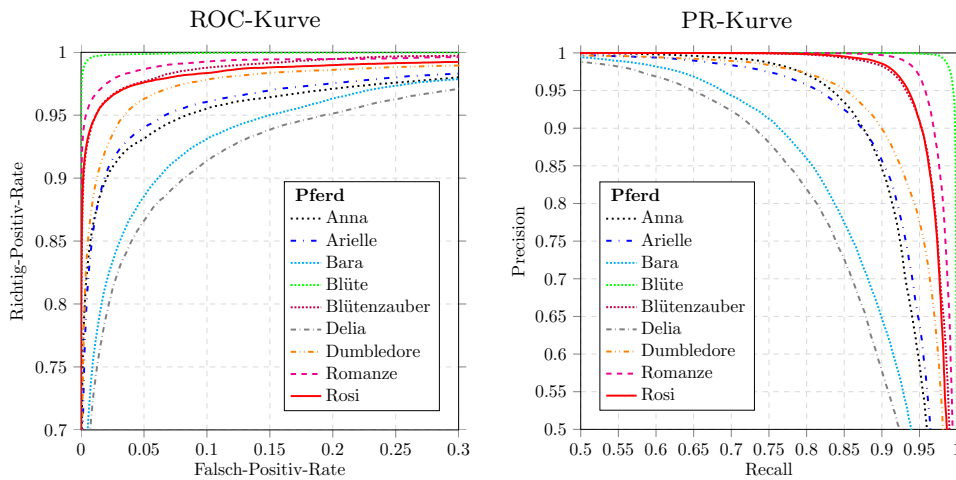
Abbildung 4.18.: CMC-Kurve der Identifikation (zweite Industriekamera) mit 36 Eigenwerten und einer 37,5% Auflösung (30×66)

darauf zu achten ist, dass die Schrittweiten zwischen den gewählten Lernanteilen nicht linear ist.

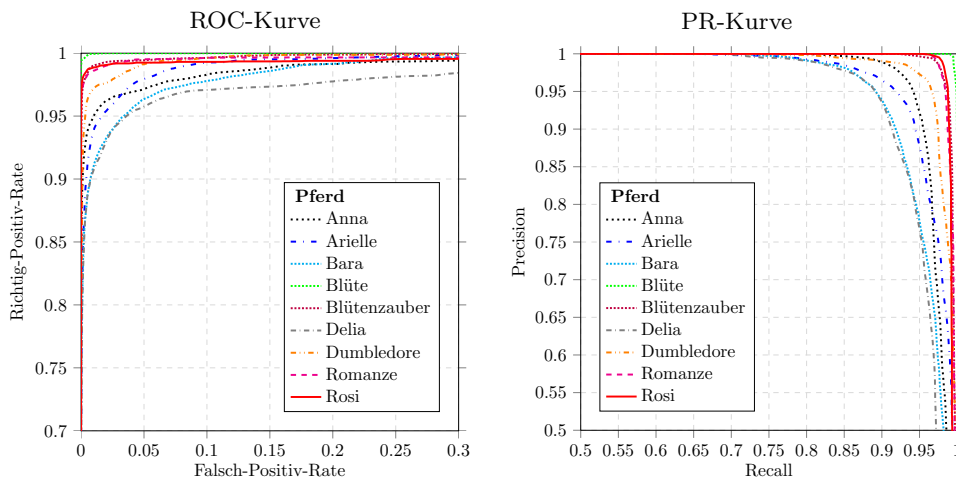
Die Abbildungen 4.19-4.21 zeigen die ROC- bzw. PR-Kurven, die in Kapitel 3.8.2 beschrieben sind, der Binärklassifikation eines jeden Pferds zu den verbleibenden angelernten Individuen in der Datenbank. Dabei bleibt die Mehrklassen-Entscheidung, die durch d_{min}^2 (siehe Kapitel 3.6.2) bestimmt wird, unberücksichtigt, sodass für jedes Individuum der Datenbank eine Distanz $d^2(\Omega', \hat{\Omega}_i')$ nach Gleichung 3.105 aus Kapitel 3.6.2 zu den Einträgen der Datenbank $\hat{\Omega}_i'$ beim Erstellen der Kurven berücksichtigt wird. Dies entspricht der Verifikation (1 : 1 Abbildung), wobei die Details dieser Auswertung in Kapitel 3.7.7 erläutert sind. Es gilt jeweils das in der Legende der Abbildungen 4.19-4.21 angegebene Pferd als positive Klasse **p** und die verbleibenden Tiere als negative Klasse **n** bei der in Kapitel 3.8.2 beschriebenen binären Klassifikation.

Die Abbildungen 4.19-4.21 sind jeweils mit unterschiedlichem Anteil von Lerndaten erzeugt worden. Abbildungen 4.19a führt die ROC- und PR-Kurven für einen Lernanteil von 5% der Datensätze, Abbildungen 4.19b den für 25% Lernanteil.

4.6. Ergebnisse der Identifizierung individueller Pferde



(a) 10 Lerndatensätze pro Pferd (5%) [36 Eigenwerte bei 37,5% Auflösung (30×66)]



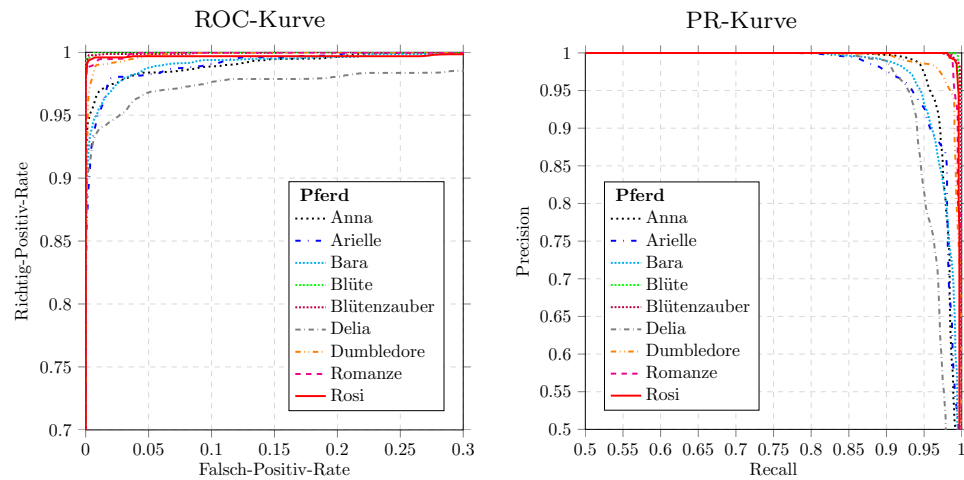
(b) 50 Lerndatensätze pro Pferd (25%) [36 Eigenwerte bei 37,5% Auflösung (30×66)]

Abbildung 4.19.: ROC- und PR-Kurven der Binärklassifikation zu jedem Individuum (2. Industriekamera, 5 und 25%)

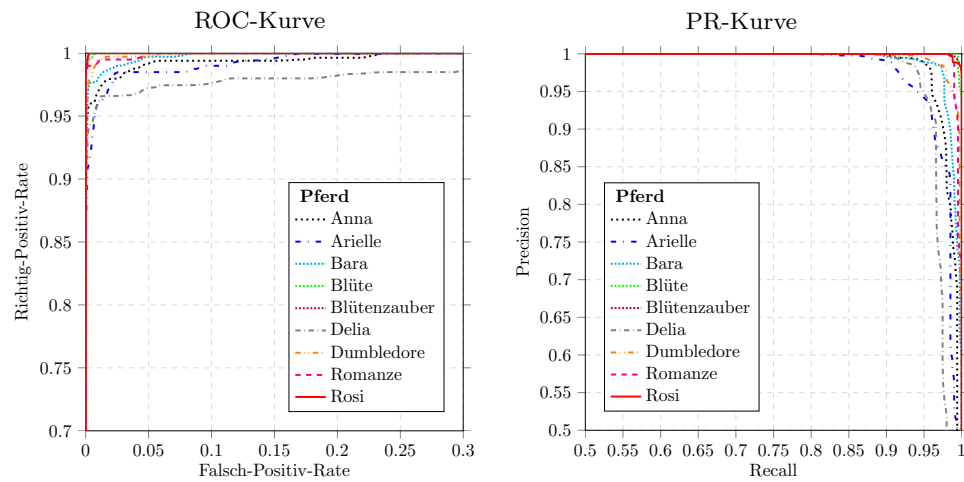
In den Abbildungen 4.20 a bzw. b werden die Kurven in denen 50% (4.20a) bzw. 75% (4.20b) Anteil der Lerndaten eingesetzt wurden gezeigt.

Die Abbildung 4.21a zeigt die ROC- und PR-Kurven für einen Lernanteil von 97,5%. Bei den abgebildeten Kurven aus 4.19-4.21 ist darauf zu achten, dass die Richtig-Positiv-Rate erst ab 0,7 und die Falsch-Positiv-Rate nur bis 0,3 angegeben ist, wodurch Details zwischen den Kurven besser zu erkennen sind. Über die einzelnen Abbildungen 4.19-4.21 von 5% Lernanteil zu 97,5% lässt sich, ähnlich wie zuvor in den Abbildungen 4.16 bzw. 4.17 und 4.18, erkennen, dass auch für die in den ROC- bzw. PR-Kurven beschriebene Binärklassifikation die Ergebnisse mit steigen-

4. Ergebnisse



(a) 100 Lerndatensätze pro Pferd (50%) [36 Eigenwerte bei 37, 5% Auflösung (30×66)]



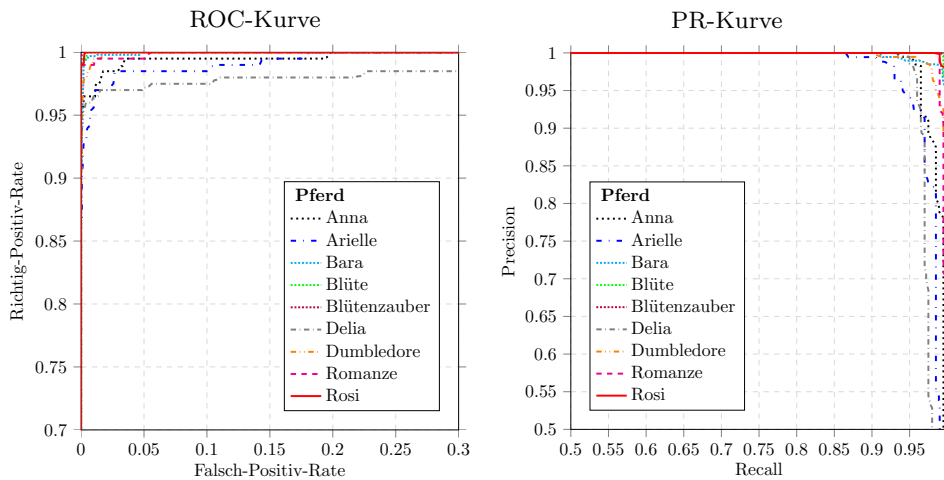
(b) 150 Lerndatensätze pro Pferd (75%) [36 Eigenwerte bei 37, 5% Auflösung (30×66)]

Abbildung 4.20.: ROC- und PR-Kurven der Binärklassifikation zu jedem Individuum (2. Industriekamera, 50 und 75%)

dem Lernanteil besser werden. Jedoch zeigt sich, im Gegensatz zu den vorherigen Abbildungen 4.16-4.18 zur Multiklassenentscheidung, deutlich, dass die Resultate pro Pferd abweichend sind. Es lässt sich erkennen, dass beispielsweise die ROC-Kurven von „Delia“ für alle Lernanteile deutlich unterhalb der ROC-Kurven der restlichen Tiere liegen.

Um zusätzlich die Unterschiede der Identifikation (mehr-Klassen Entscheidung) aus Kapitel 3.6.2 zwischen den Tieren darzustellen, wurden die Konfusionsmatrizen (siehe Tabellen 4.25-4.29) entsprechend der Beschreibung aus Kapitel 3.8.3 erstellt. Anstelle der absoluten Werte, wie sie in Abbildung 3.73 (Kapitel 3.8.3) aufgeführt

4.6. Ergebnisse der Identifizierung individueller Pferde



(a) 195 Lerndatensätze pro Pferd (97,5%) [36 Eigenwerte bei 37,5% Auflösung (30×66)]

Abbildung 4.21.: ROC- und PR-Kurven der Binärklassifikation zu jedem Individuum (2. Industriekamera, 97,5%)

sind, wurden prozentuale Werte in Abhängigkeit der ebenfalls angegebenen Testdatensätze je Pferd in den Tabellen 4.25-4.29 angegeben. Die Anzahl der Testdatensätze ergibt sich pro Tier aus der durch die Lerndaten verbleibenden Datensätze ($200 \cdot (100\% - \text{Lerndatenanteil})$) und der durch die in Kapitel 3.7.7 beschriebene *Kreuzvalidierung* 200fachen Wiederholung. Die Konfusionsmatrizen aus den Tabellen 4.25-4.29 führen somit die prozentual angegebenen Richtig-Positiv Entscheidungen auf der Diagonalen und die Fehlzuordnungen außerhalb der Diagonalen (vgl. Kapitel 3.8.3). Das arithmetische Mittel der Diagonalelemente entspricht dabei der Treffergenauigkeit (ACC) nach Gleichung 3.136 aus Kapitel 3.8.3. Die so bestimmte Treffergenauigkeit (ACC) entspricht bei gleichem Lernanteil dem Wert des ersten Ranges aus der CMC-Kurve in Abbildung 4.18 oder aber den Punkten auf der Kurve aus Abbildung 4.16 bzw. 4.17, die mit 36 Eigenwerten und einer Auflösung von 30×66 (37,5%) bestimmt wurden.

Tabelle 4.25 stellt die Konfusionsmatrix mit einem Lernanteil von 5% dar. Selbst mit einem recht geringen Anteil an Lerndaten werden im schlechtesten Fall bereits 85,3% („Delia“) der Testdaten korrekt erkannt (Einträge auf der Diagonalen). Mit einem geringen Lernanteil zeigen sich am deutlichsten die Fehlentscheidungen für die einzelnen Tiere. Es lässt sich erkennen, dass Ähnlichkeiten zwischen den Pferden das Unterscheiden der Tiere und die Identifikation des Individuums beeinträchtigen. Das die Tiere „Bara“, „Delia“ und „Rosi“, bedingt durch ihre vergleichbaren Zeichnungen, Ähnlichkeiten aufweisen, die auch in der Tabelle 4.25 vermehrt zu Fehlentscheidungen führen, ist bereits aus der Abbildung 4.15 ersichtlich. In Tabelle 4.25 werden

4. Ergebnisse

die Bilder von „Anna“ jedoch häufiger dem Pferd „Bara“ oder „Rosi“ zugeordnet als dem Pferd „Romanze“, obwohl sich „Anna“ und „Romanze“ nach Abbildung 4.15 ähnlicher sehen (beide Tiere sind Schimmel). Entsprechendes gilt auch für die Bilder von „Bara“, welche nach Tabelle 4.25 mit 1,2 % erstaunlich häufig dem Pferd „Anna“ zugeordnet wird.

Tabelle 4.25.: Konfusionsmatrix zur Identifikation zwischen den Individuen #1
36 Eigenwerte, Auflösung 30×66 (37,5 %), 5,0 % Lerndaten
 38.000 Testdatensätze

Prognose	Tatsächliche Klasse								
	Anna	Arielle	Bara	Blüte	Blütenzauber	Delia	Dumbledore	Romanze	Rosi
Anna	93,9 %	0,9 %	1,2 %	0,0 %	0,2 %	0,9 %	0,8 %	1,6 %	0,6 %
Arielle	-	91,6 %	-	0,0 %	-	-	1,2 %	-	-
Bara	1,7 %	0,4 %	89,1 %	0,1 %	0,1 %	8,4 %	2,4 %	0,1 %	0,9 %
Blüte	-	0,3 %	-	99,0 %	-	0,0 %	0,0 %	-	-
Blütenzauber	0,5 %	0,3 %	0,2 %	0,2 %	97,1 %	0,9 %	0,7 %	0,3 %	0,1 %
Delia	0,4 %	0,4 %	2,8 %	0,5 %	0,3 %	85,3 %	0,4 %	0,0 %	0,2 %
Dumbledore	-	2,2 %	-	-	-	-	91,8 %	-	-
Romanze	0,2 %	3,8 %	-	-	0,0 %	0,1 %	2,2 %	97,5 %	-
Rosi	3,3 %	0,1 %	6,7 %	0,2 %	2,3 %	4,3 %	0,5 %	0,6 %	98,1 %

Beim Vergleich der in Kapitel 3.6.1 beschriebenen Bilder mit normalisierten Grauwerten, so wird die Zuordnung der Bilder von „Anna“ zum Pferd „Bara“ und umgekehrt ersichtlich. In Abbildung 4.22a-4.22c sind Beispielbilder Γ_{Bara} , Γ_{Anna} , $\Gamma_{Romanze}$ der Pferde „Bara“, „Anna“ und „Romanze“ mit normalisierten Grauwerten gezeigt. Die Abbildung 4.22d zeigt die absolute Differenz der Bilder Γ_{Anna} und Γ_{Bara} . Abbildung 4.22e zeigt entsprechend die absolute Differenz der Bilder Γ_{Anna} und $\Gamma_{Romanze}$. Obwohl „Anna“ und „Romanze“ beides Schimmel sind und in Abbildung 4.15 deutlich ähnlicher wirken, zeigen die Differenzen der Bilder aus Abbildung 4.22d und 4.22e deutlich, dass der Unterschied zwischen den Bildern Γ_{Anna} und $\Gamma_{Romanze}$ (Abbildung 4.22e) größer ist als zwischen den Bildern Γ_{Anna} und Γ_{Bara} (Abbildung 4.22d).

Tabelle 4.26 zeigt die mehr-Klassen Entscheidung mit gleichen Randbedingungen (16 Eigenwerte und 37,5 % Auflösung 30×66), wie sie in Tabelle 4.25 gewählt wurden, jedoch mit einem höheren Lernanteil von 12,5 % (25 Bilder pro Pferd). Der prozentual angegebene Anteil der korrekten Zuordnung eines Pferdes ist auf der Diagonalen der Tabelle 4.26 im Vergleich zu Tabelle 4.25 deutlich gestiegen (arithmetisches Mittel der Steigerung 3,3 %, maximale Steigerung 6,6 %). Ein entsprechender Anstieg (arithmetisches Mittel 1,5 %, maximaler Anstieg 3,5 %) ist nochmals beim Verwen-

4.6. Ergebnisse der Identifizierung individueller Pferde

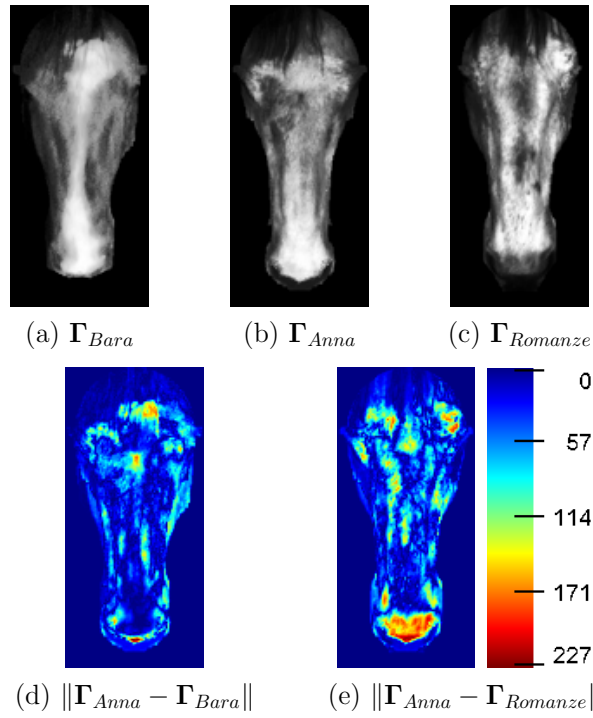


Abbildung 4.22.: Unterschied in normalisierten Grauwertbildern

den von 50 statt 25 Bildern pro Pferd (25 % Lernanteil) zum Anlernen im Vergleich zu den Ergebnissen aus Tabelle 4.26 in Tabelle 4.27 zu verzeichnen.

Deutlich geringer fallen die Unterschiede zwischen den Tabellen 4.28 und 4.29 aus, die mit einem Lernanteil von 50 % bzw. 95 % (100 bzw. 190 Bilder) erzeugt wurden. Die Steigung der Diagonalelemente beträgt zwischen den Tabellen 4.28 und 4.29 im arithmetischen Mittel lediglich 0,3 %, wobei ein maximaler Anstieg von 1 % beim Pferd „Delia“ zu verzeichnen ist. Die niedrigste Richtig-Positiv-Rate erlangt, wie zuvor in den ROC- bzw. PR-Kurven der Abbildungen 4.19-4.21 zur Verifikation, das Pferd „Delia“ mit 97,8 % bei einem Lernanteil von 95 % der Daten bzw. 96,8 % bei einem Lernanteil von 50 %. Wohingegen die restlichen Tiere bereits ab 10 Bildern zum Anlernen (5 % Lernanteil) mit einer Treffergenauigkeit (ACC) (arithmetisches Mittel der Diagonalelemente) von 94,8 % vorhergesagt werden können bzw. ab 50 Bildern (25 %) die Treffergenauigkeit bereits bei 98,8 % liegt.

4. Ergebnisse

Tabelle 4.26.: Konfusionsmatrix zur Identifikation zwischen den Individuen #2

36 Eigenwerte , Auflösung 30×66 (37,5 %), 12,5 % Lerndaten 35.000 Testdatensätze									
Prognose	Tatsächliche Klasse								
	Anna	Arielle	Bara	Blüte	Blütenzauber	Delia	Dumbledore	Romanze	Rosi
Anna	97,7 %	0,5 %	0,6 %	-	0,2 %	0,5 %	0,4 %	0,6 %	0,2 %
Arielle	-	95,9 %	-	-	-	-	0,6 %	0,0 %	-
Bara	0,8 %	0,2 %	93,9 %	0,0 %	0,0 %	5,1 %	0,6 %	0,0 %	0,6 %
Blüte	-	0,0 %	-	99,9 %	-	0,0 %	-	-	-
Blütenzauber	0,3 %	0,1 %	0,1 %	-	98,6 %	0,5 %	0,3 %	0,2 %	0,1 %
Delia	0,3 %	0,2 %	1,9 %	0,1 %	0,3 %	91,9 %	0,0 %	-	0,0 %
Dumbledore	-	0,9 %	-	-	-	-	97,2 %	-	-
Romanze	0,1 %	2,2 %	-	-	-	0,1 %	0,8 %	98,9 %	-
Rosi	0,9 %	-	3,5 %	0,0 %	0,9 %	2,0 %	0,1 %	0,2 %	99,1 %

Tabelle 4.27.: Konfusionsmatrix zur Identifikation zwischen den Individuen #3

36 Eigenwerte , Auflösung 30×66 (37,5 %), 25,0 % Lerndaten 30.000 Testdatensätze									
Prognose	Tatsächliche Klasse								
	Anna	Arielle	Bara	Blüte	Blütenzauber	Delia	Dumbledore	Romanze	Rosi
Anna	98,7 %	0,3 %	0,2 %	-	0,1 %	0,3 %	0,2 %	0,5 %	0,1 %
Arielle	-	98,1 %	-	-	-	-	0,5 %	-	-
Bara	0,4 %	0,0 %	97,0 %	-	-	3,2 %	0,2 %	-	0,6 %
Blüte	-	-	-	100,0 %	-	-	-	-	-
Blütenzauber	0,3 %	-	-	-	99,5 %	0,3 %	0,1 %	0,1 %	0,0 %
Delia	0,2 %	0,0 %	1,2 %	-	0,2 %	95,4 %	-	-	-
Dumbledore	-	0,4 %	-	-	-	-	98,8 %	-	-
Romanze	0,1 %	1,3 %	-	-	-	0,2 %	0,2 %	99,3 %	-
Rosi	0,2 %	-	1,6 %	-	0,3 %	0,6 %	-	0,1 %	99,3 %

4.6. Ergebnisse der Identifizierung individueller Pferde

Tabelle 4.28.: Konfusionsmatrix zur Identifikation zwischen den Individuen #4

36 Eigenwerte , Auflösung 30×66 (37,5 %), 50,0 % Lerndaten 20.000 Testdatensätze									
Prognose	Tatsächliche Klasse								
	Anna	Arielle	Bara	Blüte	Blütenzauber	Delia	Dumbledore	Romanze	Rosi
Anna	99,6 %	0,2 %	0,0 %	-	0,1 %	-	0,0 %	-	-
Arielle	-	99,0 %	-	-	-	-	0,5 %	-	-
Bara	-	-	98,8 %	-	-	2,2 %	-	-	0,3 %
Blüte	-	-	-	100,0 %	-	-	-	-	-
Blütenzauber	0,3 %	-	-	-	99,9 %	0,3 %	-	0,0 %	-
Delia	0,2 %	-	0,4 %	-	-	96,8 %	-	-	-
Dumbledore	-	-	-	-	-	-	99,4 %	-	-
Romanze	-	0,7 %	-	-	-	0,2 %	0,1 %	100,0 %	-
Rosi	-	-	0,8 %	-	-	0,5 %	-	-	99,7 %

Tabelle 4.29.: Konfusionsmatrix zur Identifikation zwischen den Individuen #5

36 Eigenwerte , Auflösung 30×66 (37,5 %), 95,0 % Lerndaten 2.000 Testdatensätze									
Prognose	Tatsächliche Klasse								
	Anna	Arielle	Bara	Blüte	Blütenzauber	Delia	Dumbledore	Romanze	Rosi
Anna	100,0 %	-	-	-	-	-	-	-	-
Arielle	-	99,0 %	-	-	-	-	0,5 %	-	-
Bara	-	-	99,5 %	-	-	1,2 %	-	-	-
Blüte	-	-	-	100,0 %	-	-	-	-	-
Blütenzauber	-	-	-	-	100,0 %	-	-	-	-
Delia	-	-	-	-	-	97,8 %	-	-	-
Dumbledore	-	-	-	-	-	-	99,5 %	-	-
Romanze	-	1,0 %	-	-	-	0,5 %	-	100,0 %	-
Rosi	-	-	0,5 %	-	-	0,5 %	-	-	100,0 %

5. Diskussion und Fazit

5.1. Aufnahme von 2D/3D Daten in der Versuchsstation

5.1.1. Aufnahmeeinheit und Messanordnung im Versuchsstall

Nur wenige Arbeiten befassen sich mit der *Detektion* (vgl. Kapitel 2.2.4) von landwirtschaftlichen Nutztieren aus einer Front- oder Seitenansicht (z.B. [Kim u. Jung 2013; van Hertem u. a. 2013]). In [Cangar u. a. 2008] wird ein Aufbau vorgestellt mit dem das Verhalten von großen Tieren (Kühe) untersucht und aufgezeichnet wird. Dabei wird eine Draufsicht auf die Tiere mit den Kameras erfasst, eine zweite Kamera nimmt die Seitenansicht der Tiere lediglich für die manuelle Auswertung auf. Cangar u. a. [2008] beschreibt, dass ein solcher Aufbau es erlaubt einzelne Bereiche gut abzudecken. Um eine entsprechende Abdeckung (ebenfalls bedingt durch den Öffnungswinkel der Kamera) zu ermöglichen, sollten die Kameras nach [Cangar u. a. 2008] jedoch recht hoch (5 m) angebracht werden. Dies wird in der Praxis durch die Bauhöhe der Ställe eingeschränkt, wobei die Wahl eines starken Weitwinkels, sollte der Arbeitsabstand verkleinert werden, zu hoher Verzeichnung und niedrigeren Auflösungen in Teilbereichen der Bilder führt. Der in dieser Arbeit gewählte Ansatz der Front- bzw. Seitenaufnahmen ist nach van Hertem u. a. [2013] aufgrund der unterschiedlichen Fellzeichnungen der Tiere (z.B. Rinder, Pferde) und der ungleichmässigen Hintergründe eine Herausforderung zur automatischen *Detektion*. Diesen Herausforderungen wurde durch den Einsatz einer Tiefenkamera, wie sie in Kapitel 3.3 beschrieben ist, zum Erfassen der 2D- und 3D-Bilddaten begegnet. Der Ansatz der Front- und Seitenaufnahme bietet hingegen im Vergleich zur Aufnahme einer Draufsicht einen deutlich flexibleren Einsatz. So sind in freien Bereichen (keine Verdeckung) wenig bauliche Einschränkungen zu erwarten (keine Beschränkung durch Bauhöhe), so dass Aufnahmen auch über eine größere Distanz möglich sind, welche ausschließlich durch die Eigenschaften der Kamera (Auflösung bzw. bei 3D Kameras deren Arbeitsbereich) eingeschränkt wird. Erlaubt es die Kamera, so ist die Position der Tiere im Raum auch über einen größeren Bereich hinweg feststellbar. Natürlich ist für die Unterscheidung von Tieren vorrangig entscheidend, wo sich die Merkmale am Tier befinden, um die Kameras dem Bedarf entsprechend zu positionieren. So wird die Unterscheidung anhand der Köpfe, wie sie in dieser Dissertation untersucht wurde, kaum mit Aufnahmen einer Kamera mit Draufsicht durchführbar sein, da diese den zu untersuchenden Bereich des Kopfes nicht ausreichend abdecken. Bei einer Gruppe von Tieren kann es bei seitlichen Aufnahmen, ohne eine Vereinzelung der Tiere z.B. durch eine wie in Kapitel 3.2 beschriebenen Futterstation, zu Verdeckungen der Tiere untereinander kommen. Zudem ist die Technik (Kamera)

5. Diskussion und Fazit

durch entsprechende Vorkehrungen (in dieser Arbeit durch V-Ausschnitt, siehe Kapitel 3.2) vor den Tieren zu schützen. Dies dient gleichzeitig dem Schutz der Tiere, die sich an den Gerätschaften verletzen könnten.

Alternativ ist das Anbringen der Kameras unterhalb der Decke mit einer diagonalen Ausrichtung auf den zu untersuchenden Bereich vermutlich für den praktischen Einsatz die einfachste Möglichkeit, die eingesetzte Technik außerhalb des durch die Tiere erreichbaren Bereiches zu installieren und eine ausreichende Distanz sowie Abdeckung zu gewährleisten. Mit einer entsprechenden Ausrichtung der Kamera lässt sich der Arbeitsabstand ohne einen größeren Flächenbedarf vergrößern, so dass auch Kameras mit geringerem Öffnungswinkel (mehr Auswahl auch in Bezug auf die Tiefenkameras) eingesetzt werden können oder aber der Abstand in horizontaler Richtung verkleinert werden kann. Die Ergebnisse aus Kapitel 4.3 stellen ohnehin dar, dass die Pferde mit leicht gehobenem Kopf an die Aufnahmeeinheit getreten sind. So liegt der Masseschwerpunkt des Winkels θ im Histogramm aus Abbildung 4.5 (Kapitel 4.3) für die Orientierung des Pferdekopfes ohne Selbstverdeckung bei $19,1^\circ$. Damit wäre anzunehmen, dass die Kamera im besten Fall in einem Winkel von ca. 70° von oben auf den Pferdekopf gerichtet sein sollte. Die in Kapitel 4.3 beschriebene Orientierung der Köpfe bezieht sich jedoch auf den in Kapitel 3.2 beschriebenen Versuchsstall, in dem die Kopfhaltung der Tiere durch die baulichen Gegebenheiten (erhöhter Trog oder V-Ausschnitt, siehe Abbildung 3.3 aus Kapitel 3.2) beeinflusst sein kann.

Fazit

Der in der vorliegenden Forschungsarbeit aufgezeigte Weg hin zur automatischen Identifikation von Pferden anhand der biometrischen Merkmale im Gesicht der Tiere ist auf ein Aufnahmesystem angewiesen. Das benötigte Bilderfassungssystem muss in der Lage sein gleichzeitig die Tiefeninformationen einer Szene mit den Grauwertdaten einer Standard-Kamera zu erfassen. Ein entsprechendes System, welches die Aufnahmen der Tiere unter realen Bedingungen (im Versuchsstall) erfasst, konnte im Zuge dieser Arbeit aufgebaut werden. Durch den beschriebenen Aufbau der Versuchsstation mit V-Ausschnitt konnten Tier und Technik voneinander getrennt und damit mögliche Verletzungen der Pferde oder das Beeinflussen der Technik durch die Tiere verhindert werden.

Das Auslösen der Aufnahme konnte durch die Auswertung der Tiefendaten zuverlässig ohne weitere Sensoren (z.B. Lichtschranke) neben dem beschriebenen Bilderfassungssystem durchgeführt werden.

5.1.2. Kamerakalibrierung und Tiefenmessungen

Die Ergebnisse der *Tiefenmessung 1* aus Kapitel 4.2 sind vergleichbar mit den durch Khoshelham [2011] vorgestellten Ergebnissen, bei denen ebenfalls die Fehler einer in die Tiefendaten eingepassten Ebene in unterschiedlicher Entfernung zur Kamera ausgewertet wurden. Die Ergebnisse aus Kapitel 4.2 sind entgegen der Darstellung in [Khoshelham 2011] als Boxplots (siehe Kapitel 3.8.1) dargestellt, um mögliche Verschiebungen oder systemrelevante Fehler (konnten nicht festgestellt werden) besser beurteilen zu können. Die Standardabweichung, die in dieser Arbeit nicht dargestellt wurde, entspricht dabei über die jeweiligen Distanzen (die in Kapitel 4.2 lediglich bis $3,8\text{ m}$ gemessen wurden) der Standardabweichung aus [Khoshelham 2011]. Es ist zu beachten, dass ca. 68 % der Daten innerhalb der durch die Standardabweichung angegebenen Schranken liegen. Die Ergebnisse aus Kapitel 4.2 sind vergleichbar mit denen aus [Khoshelham 2011], obwohl durch Khoshelham [2011] eine abweichende Kamerakalibrierung zu der in Kapitel 3.3.2 beschriebenen Kamerakalibrierung nach Herrera u. a. [2011] eingesetzt wurde. Aus den Ergebnissen der *Tiefenmessung 1* lässt sich die These von Khoshelham [2011] bestätigen, dass die Fehler der Tiefenmessung quadratisch mit der Entfernung zunimmt. Die Tiefendaten zur Auswertung der Detektion aus Kapitel 3.7.4 als auch die *Orientierung der Pferdeköpfe* aus Kapitel 4.3 wurden nach der Beschreibung aus Kapitel 3.7.4 auf Pferdeköpfe mit einer maximalen Distanz von 3 m zur Kamera beschränkt. Wie durch Khoshelham [2011] beschrieben, fällt die Tiefenmessung oberhalb von 3 m deutlich schlechter aus. Die Absolutfehler innerhalb der 3 m liegen nach den Ergebnissen aus Kapitel 4.2 hauptsächlich (90 %) unterhalb von 1,7 % der jeweiligen Distanz.

Die Messungen der Linienlängen aus Unterkapitel *Tiefenmessung 2* haben geringere Fehler aufzuweisen als die der reinen Tiefenmessung aus Unterkapitel *Tiefenmessung 1* des Kapitels 4.2. Wobei die Messfehler in Kapitel 4.2 der horizontalen Linien aus Abbildung 4.2 größer waren als die der vertikalen Linien aus Abbildung 4.3. Wird die Abbildung 4.2 der horizontalen Linienmessfehler nur für die Kalibrierobjekte #2 und #3 der unteren Reihe aus Abbildung 3.56 des Kapitels 3.7.1 erstellt, so ergeben sich die Fehler wie sie in Abbildung 5.1 in Blau dargestellt sind. Zum Vergleich sind die Fehler der vertikalen Linien für die entsprechenden Kalibrierobjekte in der Abbildung 5.1 in Grau eingezeichnet. Die Messfehler der horizontalen und vertikalen Linienlängen unterscheiden sich in der Abbildung 5.1 nur noch geringfügig. Die Unterschiede in den Messfehlern der horizontalen bzw. vertikalen Linienlänge sind damit auf die Orientierung der Kalibrierobjekte zur Kamera zurückzuführen. Der Fehler ist, wie auch in [Khoshelham 2011] beobachtet wurde, größer, sobald der Einfallswinkel der Lichtstrahlen des Projektors (strukturiertes Licht siehe Kapitel 3.3) auf die zu messende Oberfläche größer wird.

5. Diskussion und Fazit

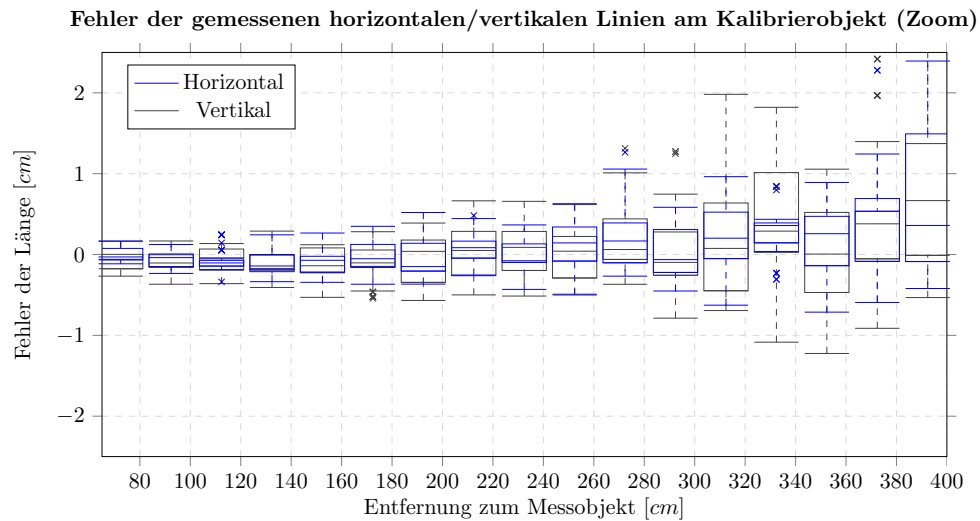


Abbildung 5.1.: Längenfehler der horizontalen (blau) bzw. vertikalen (grau) Linien des Kalibrierobjekts (#2 und #3) in unterschiedlichen Entfernungen

Fazit

Die Ergebnisse aus Kapitel 4.2 geben einen Einblick auf die in der jeweiligen Entfernung zu erwartenden Messfehler der in Kapitel 3.3 beschriebenen Tiefenkamera. Durch den Einsatz einer Kalibrierung, die zu der in Kapitel 3.3.2 verwendeten Kamerakalibrierung zusätzlich die Verzerrung der Infrarotkamera und damit der Tiefenmesswerte berücksichtigt, wie durch Khoshelham u. Elberink [2012] oder Herrera u. a. [2012] vorgestellt, kann die Messgenauigkeit weiter verbessert werden. Zudem sollte die Kalibrierung zukünftig wie in [Herrera u. a. 2012] mit einer Fehlerfunktion über alle verwendeten Kameras erstellt werden. Denkbar ist auch die Tiefeninformationen der Kinect mit Aufnahmen mehrerer Kameras zu kombinieren, um so ähnlich der Beschreibung von Smisek u. a. [2011] 3D Rekonstruktionen mit größerem Detail erstellen zu können.

5.2. Orientierung der Pferdeköpfe

Die Ergebnisse aus Kapitel 4.3 wurden im Zusammenhang mit den Vergleichsdaten (siehe Kapitel 3.7.4) erstellt. Es wird dargestellt in welchen Bereichen die Orientierung der Pferdeköpfe, unter Berücksichtigung des Messaufbaus aus Kapitel 3.2, zu Selbstverdeckung führt. Abließend kann eine Ellipse in Abhängigkeit der Orientierungswinkel (ψ und θ) der Pferdeköpfe bestimmt werden, in denen keine Selbstverdeckung zu erwarten ist. Nur entsprechende Bilddaten mit komplett abgebildeten Köpfen und ohne eine Selbstverdeckung kommen für eine Identifikation nach Kapitel 3.6 in Frage. Im Zuge einer automatischen Bildverarbeitungskette zur Identifika-

tion (siehe Abbildung 1.1 aus Kapitel 1.1) sind diese Informationen erst nach einer *Detektion* (Kapitel 3.4) sowie der Posenbestimmung der Köpfe (Kapitel 3.5.1 bzw. 3.5.2) zur Auswahl der Bilder für eine Identifikation (Kapitel 3.6) nutzbar.

Die Verteilung in der Orientierung der Pferdeköpfe zur Kamera ist dahingehend relevant, dass die Kamera eine möglichst große Anzahl an verwertbaren Daten erfasst. So können die Informationen der Histogramme (Abbildung 4.5 und 4.4) aus Kapitel 4.3 dazu genutzt werden, die Kamera neu zu positionieren und die Anzahl der nutzbaren Bilddaten hoch zu halten (vgl. Diskussion aus Kapitel 5.1.1 zur Kamerapositionierung).

In den Histogrammen aus Abbildung 4.4 und 4.5 sind lediglich Datensätze eingetragen, bei denen alle Marker (Nüstern und Augen) gesetzt wurden. Die Winkelorientierungen der Datensätze mit unvollständiger Markierung sollte jedoch deutlich höher sein als die Winkel innerhalb der in Kapitel 4.3 beschriebenen Ellipse. Somit ist davon auszugehen, dass die Mehrheit der Pferdeköpfe ohne vollständig sichtbare Markierungen, zumindest im Fall einer Selbstverdeckung, ebenfalls ignoriert werden. Die im Kapitel 4.3 angegebene Treffergenauigkeit liegt daher, aufgrund der berücksichtigten Datensätze, unterhalb der voraussichtlich deutlich höher anzusetzenden Treffergenauigkeit. Anzumerken ist, dass das manuelle Markieren der Daten auf Selbstverdeckung auf den 2D Bilddaten erfolgte. Die Daten wurden dabei auf einer Kamera markiert, welche eine andere Orientierung auf die Szene besitzt, als die Tiefenkamera, mit der letztlich die Orientierungswinkel bestimmt wurden. Vorrangig in den Bilddaten, in denen sich die Pferdeköpfe in einem Bereich des Übergangs zur Selbstverdeckung befinden, vergleichbar mit den Winkeln auf der Ellipse, ist die Markierung nur ungenau zu setzen. Werden die Winkel zuvor in die Orientierungen der jeweils übrigen Kameraansichten transformiert, wäre so eine Auswahl der, in der entsprechenden Kamera ohne Selbstverdeckung sichtbaren Daten, besser möglich.

Fazit

Sollte die Kameraposition verändert werden (z.B. unterhalb der Decke), ist zu untersuchen, ob diese die Aufnahme in Bezug auf eine quantitativ häufig festzustellende Kopfhaltung der Tiere zu sehr einschränkt. Hierzu kann die in der vorliegenden Arbeit bereits durchgeführte Auswertung lediglich Anhaltspunkte liefern, inwiefern ein Umbau der Station jedoch das Verhalten der Tiere und damit deren gewohnte Kopfhaltung beeinflusst kann nicht vorhergesagt werden.

Anhand der durchgeführten quantitativen Auswertung der Orientierung des Kopfes der Pferde während ihrer Besuche in der Futterstation konnten Bereiche bestimmt werden, in denen auf den Bilddaten mit einer Selbstverdeckung zu rechnen ist. Durch die unterschiedlichen Zeichnungen beim Pferd ist eine Rekonstruktion entsprechend verdeckter Bereiche im Gesicht der Tiere, um die Daten für eine Identifikation zu vervollständigen, nicht zu empfehlen. Eine Alternative wäre das Anlernen unter-

schiedlicher Teilbereiche des Pferdegesichts, um außerhalb der genannten Grenzen operieren zu können. Inwiefern das Auswerten von Teilbereichen die Identifikationsergebnisse beeinflusst wäre zukünftig ebenfalls zu untersuchen.

5.3. Detektion

Das Verfahren von Viola u. Jones [2004] gilt, wie bereits in der Literaturübersicht im Kapitel 2.2.2 beschrieben, als das Standardverfahren der Gesichtsdetektion beim Menschen [Zhang u. Zhang 2010] auf zweidimensionalen Bildern. Obwohl das Verfahren bereits durch Burghardt u. a. [2004a]; Burghardt u. Calic [2006a, b]; Zhang u. a. [2008]; Kozakaya u. a. [2009]; Martinez-Ortiz u. a. [2013] zur Detektion von Gesichtern unterschiedlicher Tiere eingesetzt wurde, wird es in dieser Dissertation nicht weiter verwendet. Andreopoulos u. Tsotsos [2013] beschreibt, dass die Methode von Viola u. Jones [2004] nur zur Detektion von einfachen Objekten, die sich durch eine kleine Anzahl an typischen Merkmalen auszeichnen, eingesetzt werden sollte. Andreopoulos u. Tsotsos [2013] nennt hier menschliche Gesichter und Türknöpfe. Die unterschiedlichen Zeichnungen der in dieser Dissertation untersuchten Pferde fallen nicht in diese Gruppe. Auch eine Anpassung der verwendeten Feature (HOOG - *Haar of Oriented Gradients* bzw. CoHOG - *co-occurrence histograms of oriented gradients*), wie sie durch Zhang u. a. [2008] oder Kozakaya u. a. [2009] vorgestellt wurden, wird bei den Pferden nicht zu vergleichbaren Resultaten wie bei der Detektion von Katzengesichtern führen. Die seitliche Ausrichtung der Augen bei den Pferden [Timney u. Keil 1999] erschwert es vergleichbare Strukturen im Gesicht unterschiedlicher Tiere aus zweidimensionalen Bilddaten zu extrahieren. Zudem werden zum Anlernen des Verfahrens von Viola u. Jones [2004] eine möglichst große Anzahl an Bilddaten von mehreren Tieren (in [Zhang u. a. 2008] 10.000 gelabelte Bilder) benötigt. Die im Rahmen dieser Dissertation erstellte Datenbank enthält jedoch nur Bilder von 14 Pferden, wobei nur von 9 Pferden ausreichend Bildmaterial gesammelt werden konnte (siehe Tabelle 4.1 der gesammelten Bilddaten aus Kapitel 4.1 bzw. Tabellen 4.23 und 4.24 der detektierten Pferdeköpfe aus Kapitel 4.6.1).

Im Gegensatz zur Detektion auf zweidimensionalen Bilddaten haben die unterschiedlichen Zeichnungen und Fellfarben der Pferde keinen Einfluss auf die Tiefeninformation. Die Kopfform der Pferde zeigt eine deutlich geringere Variation zwischen den Tieren, als ihre Zeichnung. Zudem ist die Trennung von Tier und Hintergrund über die Tiefeninformation deutlich einfacher und zuverlässiger, weshalb sich die Detektion in dieser Dissertation auf die Tiefeninformation der Kamera aus Kapitel 3.3 beschränkt.

In Kapitel 3.4.2 bzw. 3.4.3 wurden zwei Verfahren (*Clustern durch Hough-„Voting“* bzw. *Segmentierung durch Winkel der Ebenennormalen*) vorgestellt, um die einzelnen im Tiefenbild vorhandenen Regionen zu differenzieren bzw. zu segmentieren. Die aufgefundenen Bereiche werden anschließend mit dem in Kapitel 3.4.4 vorgestellten

Verfahren (*Detektion von Kopfkandidaten aus segmentierten Bilddaten*) dahingehend untersucht, ob es sich bei dem Bereich um einen Pferdekopf handelt. Dieses Vorgehen entspricht für beide Verfahren (Kapitel 3.4.2 bzw. 3.4.3) der in [Zhang u. a. 2013] als „Lokalisierung durch Segmentierung“ (*localization through segmentation*) bezeichneten Lokalisierungs-Strategie, welche neben den Strategien der „Lokalisierung durch Fenster-Suche“ (*localization through sub-window search*) und der „Voting-Strategie“ (*voting strategy*) aufgeführt wird. Auch für das Hough-„Voting“ aus Kapitel 3.4.2 findet die eigentliche Detektion erst mit der Klassifikation in Kapitel 3.4.4 statt. In [Zhang u. a. 2013] ist ebenfalls beschrieben, dass die durch die vorgestellten Verfahren erreichten Genauigkeiten nicht für den praktischen Einsatz ausreichen, weshalb in dieser Dissertation auf die Kopfform der Pferde spezialisierte Ansätze (besonders Kapitel 3.4.3) zum Auffinden der Pferdeköpfe beschrieben sind.

Fazit

Die Detektion der Tiere auf den erfassten Daten erfolgt rein auf den Tiefendaten, was es ermöglicht die Tiere unabhängig von ihrer Fellfarbe oder -zeichnung in den Bilddaten aufzufinden.

5.3.1. Segmentierung und Regionenmarkierung

Die Vereinzelung der Bereiche im Bild der in Kapitel 3.4 beschriebenen Verfahren (das *Clustern durch Hough-„Voting“* und die *Segmentierung durch Winkel der Ebenennormalen*) lässt sich, wie durch Silberman u. a. [2012] vorgestellt, auch zur Interpretation von Szenen ausnutzen. Das Verfahren zum *Clustern durch Hough-„Voting“* aus Kapitel 3.4.2 setzt dabei, wie auch Silberman u. a. [2012], auf die Flächen, wobei in Kapitel 3.4.2 lediglich die größte Fläche im Vordergrund extrahiert wird. Das Verfahren zur *Segmentierung durch Winkel der Ebenennormalen* aus Kapitel 3.4.3 unterteilt hingegen das gesamte Tiefenbild, wobei im Gegensatz zu der Arbeit [Silberman u. a. 2012] Löcher (Bereiche ohne Zuordnung) zwischen den aufgefundenen Regionen auftreten können und in [Silberman u. a. 2012] die Farbbildinformationen berücksichtigt werden. Zudem scheint der Ansatz von Silberman u. a. [2012] deutlich rechenintensiver zu sein, da das Verfahren aus Kapitel 3.4.3 direkt zur Regionenmarkierung (siehe Kapitel 3.4.3) genutzt wird und die Regionen nicht, wie durch Silberman u. a. [2012], zusammengeführt werden. Beim Vergleich der Verfahren aus Kapitel 3.4.2 und 3.4.3 fällt auf, dass für das Verfahren aus Kapitel 3.4.2, welches darauf ausgelegt ist durch das Hough-„Voting“ große Flächen im Bild aufzufinden, zunächst der Hintergrund zu entfernen ist (siehe Ablauf aus Abbildung 3.23 in Kapitel 3.4.2). So muss der Hintergrund für das Verfahren aus Kapitel 3.4.2 zuvor angelernt werden. Das in Kapitel 3.4.3 vorgestellte Verfahren kommt hingegen ohne diesen Schritt aus und kann das komplette Bild, mitsamt Hintergrund, anhand der zuvor berechneten Winkel aus Kapitel 3.4.1 in Einzelbereiche einteilen.

Die in Gleichung 3.111 aus Kapitel 3.7.5 erstellte Kostenfunktion ermöglicht es an

5. Diskussion und Fazit

Hand der Vergleichsdaten aus Kapitel 3.7.3 die beiden Verfahren aus Kapitel 3.4.2 und 3.4.3 zu vergleichen. Dabei hat sich bereits im Kapitel 4.4.1 das Verfahren aus Kapitel 3.4.3 als dem Verfahren aus Kapitel 3.4.2 überlegen dargestellt, welches sich durch die in Kapitel 4.4.2 beschriebenen Detektionsraten bestätigt. Das Verfahren aus Kapitel 3.4.3 kommt dabei nach der beschriebenen Kostenfunktion (aus Gleichung 3.111 in Kapitel 3.7.5) zu besseren Ergebnissen, obwohl es ohne eine Trennung von Vorder- und Hintergrund auskommt.

Die Kombination der optionalen Parameter aus Tabelle 3.6 in Kapitel 3.7.5 ergeben unterschiedliche Parametersätze für das Verfahren aus Kapitel 3.4.3. Unter diesen Parametersätzen erzielen die Parameterkombinationen $\{\alpha, t_b, T_\theta\}$ und $\{\alpha, t_b, T_\psi, T_\theta\}$ (vgl. *Parametervergleich I* aus Kapitel 4.4.1) nach der Tabelle 4.7 aus Kapitel 4.4.1 die besten Ergebnisse. Wie in Kapitel 4.4.3 beschrieben, fällt die Wahl auf den Parametersatz $\{\alpha, t_b, T_\psi, T_\theta\}$.

Obwohl die Variation der Parameter (vorrangig $T_\psi : 5,3^\circ \leq T_\psi \leq 14,7^\circ$), die jeweils für die 15 Lerndatensätze optimiert wurden (vgl. Kapitel 3.7.5), recht hoch ist, ergibt sich nach Tabelle E.4 aus Anhang E.1.1 eine vergleichsweise geringe Streuung ($72,0 \leq c_I \leq 75,7$) in den Kosten c_I (nach Gleichung 3.111 aus Kapitel 3.7.5) der Lerndatensätze. Die Streuung $\Delta 5,5$ der Kosten \hat{c}_I ($73,2 \leq \hat{c}_I \leq 77,7$) aus Tabelle E.4 in Anhang E.1.1 für die Testdatensätze des Parametersatzes $\{\alpha, t_b, T_\psi, T_\theta\}$ ist nur für den Parametersatz $\{\alpha, t_b, T_\theta\}$ in Tabelle E.2 (Anhang E.1.1) mit $\Delta 4,6$ ($74,2 \leq \hat{c}_I \leq 78,9$) geringer. Zudem zeigen die Ergebnisse der *Parameterwahl I* aus Anhang E.2.1 in den Tabellen E.26-E.28 für den Median und das arithmetische Mittel \emptyset der optimierten Parameter des Parametersatzes $\{\alpha, t_b, T_\psi, T_\theta\}$ ebenfalls eine geringe Streuung der Kosten c_I bzw. \hat{c}_I für die 15 Lern- bzw. Testdatensätze ($72,6 \leq c_I \leq 77,3$ bzw. $73,0 \leq \hat{c}_I \leq 77,0$), trotzdem keine pro Datensatz optimierten Parameter verwendet wurden. Aufgrund dieser Ergebnisse lässt sich das Verfahren aus Kapitel 3.4.3 mit dem Parametersatz $\{\alpha, t_b, T_\psi, T_\theta\}$ als robust gegenüber kleinen Parametervariationen (im Bereich der als Median bzw. arithmetischen Mittel aufgeführten Parameter aus Tabelle 4.14) bezeichnen. Die Ergebnisse ($78,7 \leq c_I \leq 86,8$ bzw. $78,4 \leq \hat{c}_I \leq 84,4$) der maximalen Parameter aus den Tabellen E.26-E.28 (Anhang E.2.1) fallen, verglichen mit den Ergebnissen ($92,4 \leq c_I \leq 98,2$ bzw. $93,6 \leq \hat{c}_I \leq 98,0$) der minimalen Parameter, deutlich besseren aus, weshalb darauf geachtet werden sollte, die Parameter nicht zu klein zu wählen.

Für einige wenige Datensätze der Tabelle 4.7 aus Kapitel 4.4.1, welche die Ergebnisse der Segmentierung nach Kapitel 3.4.3 führt, lässt sich erkennen, dass es sich bei den durch das in Kapitel 3.7.5 beschriebene Optimierungsverfahren bestimmten Ergebnissen um kein globales Minimum handelt. Dies wird durch den Vergleich der Ergebnisse in Tabelle 4.7 aus Kapitel 4.4.1 mit denen der Tabellen E.26-E.28 zur *Parameterwahl I* aus Anhang E.2.1 deutlich. So sind die Ergebnisse des Parametersatzes $\{T_\psi, T_\theta, T_\delta\}$ für die Lerndatensätze #3, #6, #8 und #10 in den Tabellen E.26-E.28 der *Parameterwahl I* (Anhang E.2.1) kleiner als die für jeden Datensatz optimierten Ergebnisse der Tabelle 4.7 (Kapitel 4.4.1). Entsprechendes

gilt für den Datensatz #2 des Parametersatzes $\{T_\psi, T_\theta\}$ und den Datensatz #1 des Parametersatzes $\{T_\psi, T_\delta\}$. Die Optimierung der Parameter aus Kapitel 3.7.5 nach der diskreten, nicht differenzierbaren Kostenfunktion (Gleichung 3.111) ist demnach nicht optimal.

Alternativ zu dem in Kapitel 3.7.5 verwendeten Generalized Pattern Search (GPS) Algorithmus [Kolda u. a. 2003] lassen sich Evolutionäre Algorithmen (EA - *evolutionary algorithms*) [Tomassini 1996] oder Verfahren der simulierten Abkühlung (SA - *simulated annealing*) [Buseti 2003] einsetzen. Die Arbeit von Abramson [2002] beschreibt den Generalized Pattern Search (GPS) Algorithmus zur Optimierung mit gemischten Variablen (kontinuierlichen und diskreten), zudem werden die zuvor genannten Verfahren (Evolutionäre Algorithmen und simulierte Abkühlung) grob erklärt und mit dem Generalized Pattern Search (GPS) Algorithmus verglichen. Die Optimierung durch die simulierte Abkühlung (SA - *simulated annealing*) wurde probenhalber auf einem der Datensätze getestet und braucht für das Auffinden des Optimums deutlich länger (auf einem Intel i7 mit 2,93 GHz single Thread: SA - 4627 Funktionsauswertungen [74,5 Stunden] gegen GPS - 367 Funktionsauswertungen [7,2 Stunden]). Zudem liegen die Kosten c_I mit den beiden Optimierungsverfahren (GPS [Kolda u. a. 2003] bzw. SA [Buseti 2003]) bestimmten Parametern für den Datensatz #1 des Parametersatzes $\{T_\psi, T_\theta\}$ mit $c_I = 75,3$ bzw. $c_I = 73,8$ recht nahe beieinander. Letztlich garantiert das Verfahren der simulierten Abkühlung (SA - *simulated annealing*), entgegen der Evolutionären Algorithmen [Tomassini 1996] oder dem Generalized Pattern Search (GPS) Algorithmus [Kolda u. a. 2003], das Erreichen des globalen Minimums nach Buseti [2003] bei einer unendlichen Laufzeit. Der Generalized Pattern Search (GPS) Algorithmus [Kolda u. a. 2003] ist hingegen effizienter und führt in der genannten Probe zu vergleichbaren Ergebnissen.

Abbildung 5.2 zeigt beispielhaft die Kosten c_I zur Parametervariation der ersten beiden Parameter α und t_b des Parametersatzes $\{\alpha, t_b, T_\psi, T_\theta\}$ für den ersten Lerndatensatz bei festen Parametern T_ψ und T_θ . Abbildung 5.2 zeigt, dass t_b im Bereich zwischen 7 und 10,5 und α zwischen 52° und 66° liegen sollte, damit für den ersten Lerndatensatz die Kosten c_I unterhalb von 80 (nach Gleichung 3.111 aus Kapitel 3.7.5) bleiben. Die Kosten steigen mit dem Parameter t_b zu den Seiten hin stark an, wohingegen sich die Kosten durch die Variation des Parameters α deutlich langsamer verändern und nur eine leichte Steigung bei größeren Werten für α ($\alpha > 66^\circ$) aufweisen ($c_I = 92,8$ bei $\alpha = 90^\circ$ und $t_b = 8$).

Abbildung 5.3 zeigt entsprechend der Abbildung 5.2 die Kosten c_I zur Parametervariation der letzten beiden Parameter T_ψ und T_θ des Parametersatzes $\{\alpha, t_b, T_\psi, T_\theta\}$ für den ersten Lerndatensatz bei festen Parametern α und t_b . Aus Abbildung 5.3 ist erkennbar, dass der Parameter T_ψ nur geringen Einfluss auf die Kosten c_I hat. Dies wurde bereits in Kapitel 4.4.1 festgestellt, da der Parametersatz $\{\alpha, t_b, T_\theta\}$ nur geringfügig schlechtere Ergebnisse im Vergleich zum verwendeten Parametersatz $\{\alpha, t_b, T_\psi, T_\theta\}$ erzielt. Wird der Parameter T_ψ größer als $6,5^\circ$ gewählt, so hat dieser nach Abbildung 5.3 nahezu keinen Einfluss auf die Kosten c_I in Bezug auf den ersten Lerndatensatz. Liegt der Parameter T_θ zudem im Bereich zwischen $3,5^\circ$ und $6,5^\circ$,

5. Diskussion und Fazit

so liegen auch die Kosten c_I für den ersten Lerndatensatz unterhalb von 85 (nach Gleichung 3.111 aus Kapitel 3.7.5).

Durch die Abbildungen 5.2 und 5.3 lässt sich bestätigen, dass das Verfahren aus Kapitel 3.4.3 mit den Parametern $\alpha = 57,6^\circ$, $t_b = 9,3$, $T_\psi = 9,1^\circ$ und $T_\theta = 5,6^\circ$ der *Parameterwahl I* aus Kapitel 4.4.3 für den ersten Lerndatensatz robust gegenüber kleinen Parametervariationen ist und zeigt weitere Schranken auf in denen die Parameter zu wählen sind. Zudem liegen die durch den Generalized Pattern Search (GPS) Algorithmus der Parameteroptimierung nach Kapitel 3.7.5 bestimmten Parameter in den Abbildungen 5.2 und 5.3 innerhalb der als Senken erkennbaren Bereiche, welches den Generalized Pattern Search (GPS) Algorithmus mit Berücksichtigung der Effizienz im Vergleich zu den Verfahren der simulierten Abkühlung (SA - *simulated annealing*) als geeignet darstellt.

Fazit

Anstatt eines sequenziellen Verfahrens, wie das von Stockman u. Shapiro [2001], welches in Kapitel 3.4.2 verwendet und auf dem die *Erweiterung der Regionenmarkierung auf Mehrkanal-Bilder* in Kapitel 3.4.3 basiert, lässt sich auch das parallele Verfahren von Cohen [1996] anpassen. Zudem kann mit dem Verfahren von Cohen [1996] das auf Seite 74 aus Kapitel 3.4.2 beschriebene Erzeugen der einzelnen Binärbilder für jede Ebene umgangen werden, da es in der Lage ist Grauwertbilder in seine Regionen zu unterteilen. Bei der Implementierung ist jedoch auf die Anzahl der während der Ausführung entstehenden Markierungen (Label) zu achten, da diese die Bittiefe des Regionen-Bilds vorgibt und die Verarbeitungsgeschwindigkeit beeinflussen kann. Der Unterschied liegt bei einer Parallelisierung der Verfahren nach Cohen [1996] in der Geschwindigkeit, führt jedoch zu gleichen Ergebnissen.

5.3.2. Detektion der Pferdeköpfe

Die eigentliche Detektion aus Kapitel 3.4.4 nutzt zur Klassifikation der Einzelbereiche einen zu den HONV (*histogram of oriented normal vector*) vergleichbaren Ansatz. In [Han u. a. 2013] bzw. [Tang u. a. 2013] wird beschrieben das die HONV (*histogram of oriented normal vector*) den HOG (*histogram of oriented gradients* - Richtungshistogramme z.B. [Dalal u. Triggs 2005]) Merkmalen, wie sie z.B. durch Zhang u. a. [2008] oder Kozakaya u. a. [2009] in abgewandelter Form verwendet werden, bei der Detektion unterschiedlicher Objekte überlegen ist.

Tabelle 4.9 aus Kapitel 4.4.1 führt die nach Gleichung 3.112 (Kapitel 3.7.5) berechneten Kosten c_{II} bzw. \hat{c}_{II} für die in Tabelle 3.7 (Kapitel 3.7.5) beschriebenen Parameterkombinationen auf. Zur Detektion aus Kapitel 3.4.4 mit der Segmentierung aus Kapitel 3.4.3 wurde, durch die in der Tabelle 4.9 (Kapitel 4.4.1) aufgeführten Kosten, der Parametersatz $\{b_\psi, b_\theta\}$ unter den beschriebenen Parameterkombinationen als der mit den besten Resultaten identifiziert. Die Ergebnisse der Detektion aus Kapitel 3.4.4 in Kombination mit dem *Clustern durch Hough-„Voting“* aus

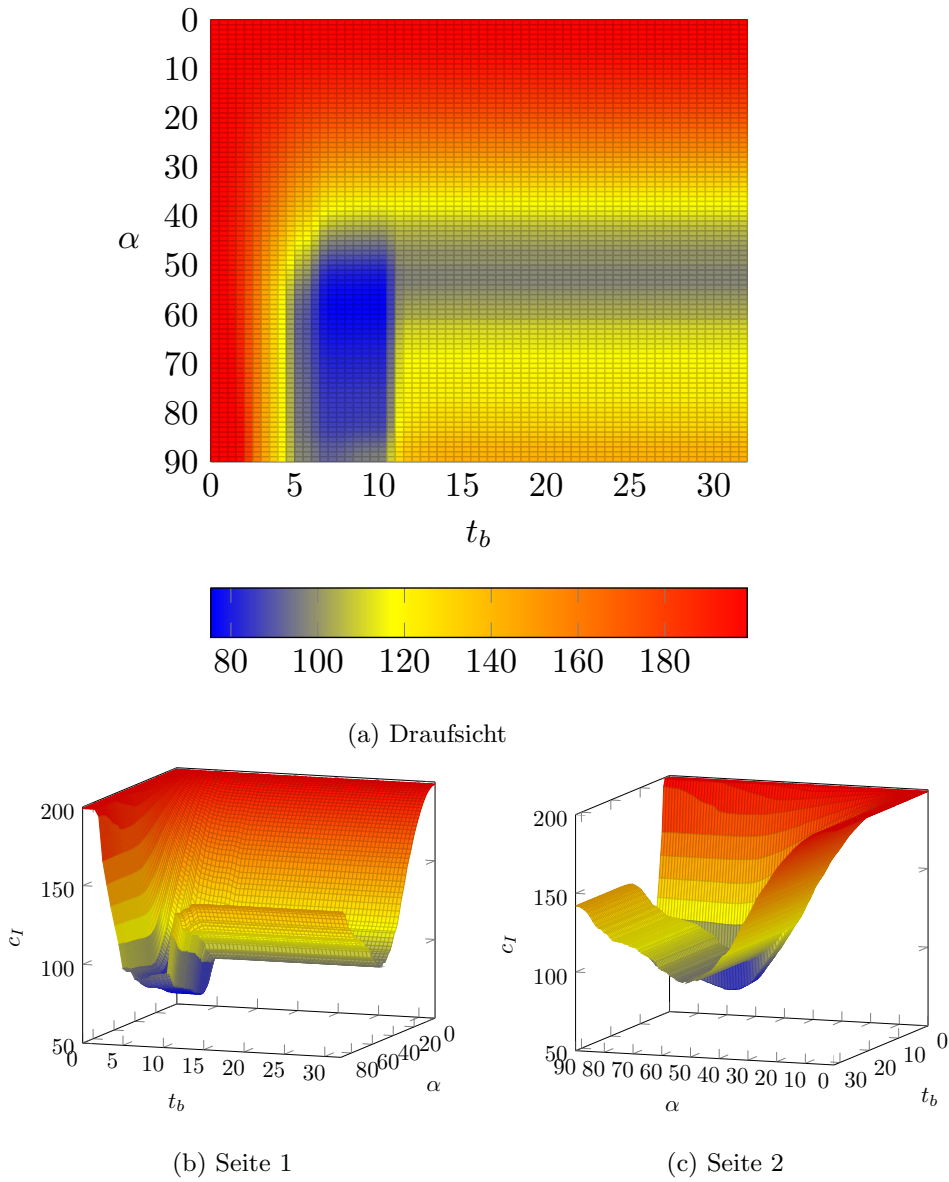
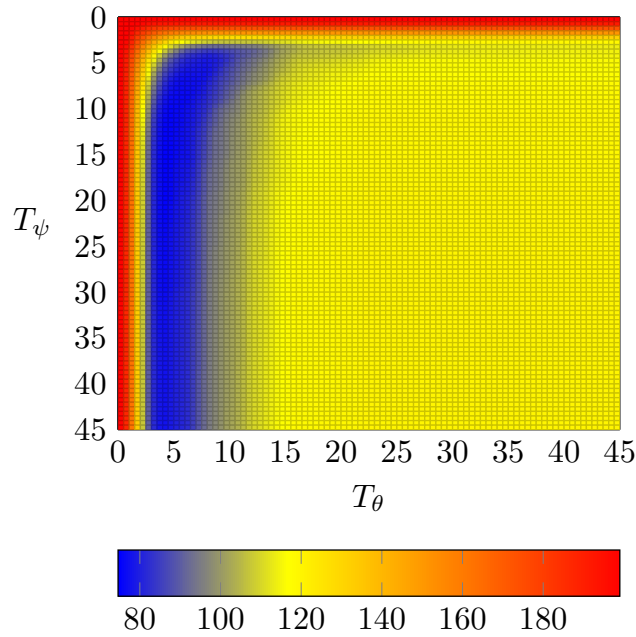


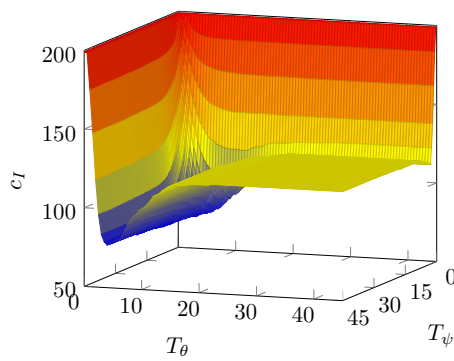
Abbildung 5.2.: Kosten zur Parametervariation der Parameter α und t_b ($T_\psi = 9.1^\circ$ und $T_\theta = 5.6^\circ$ fest) des ersten Lerndatensatzes

Kapitel 3.4.2, die in Tabelle 4.8 (Kapitel 3.7.5) aufgeführt sind, fallen, wie bereits beim Segmentieren (Kapitel 5.3.1), deutlich schlechter aus als mit dem Verfahren nach Kapitel 3.4.3.

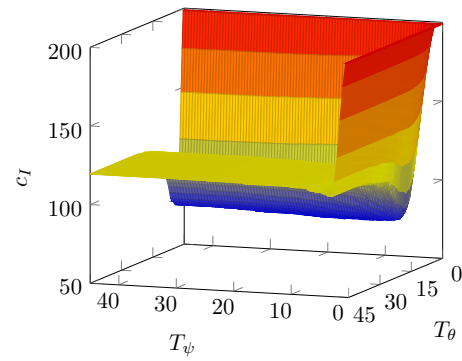
5. Diskussion und Fazit



(a) Draufsicht



(b) Seite 1



(c) Seite 2

Abbildung 5.3.: Kosten zur Parametervariation der Parameter T_ψ und T_θ ($\alpha = 57,6^\circ$ und $t_b = 9,3$ fest) des ersten Lerndatensatzes

Die Variation des nach der Kostenfunktion c_{II} optimierten Parameters b_ψ hat, nach der in Anhang E.1.2 aufgeführten Tabelle E.17, eine vergleichsweise hohe Variation zwischen den 15 Lerndatensätzen ($7,3^\circ \leq b_\psi \leq 44,9^\circ$). Wohingegen der Parameter b_θ kaum Variation zwischen den Datensätzen aufweist ($12,7^\circ \leq b_\theta \leq 14,9^\circ$). Die Kosten c_{II} nach der *Parameterwahl II* aus Kapitel 4.4.3 zum Parametersatz $\{b_\psi, b_\theta\}$, welche in den Tabellen E.29-E.31 aus Anhang E.2.2 zu finden sind und im Bereich zwischen

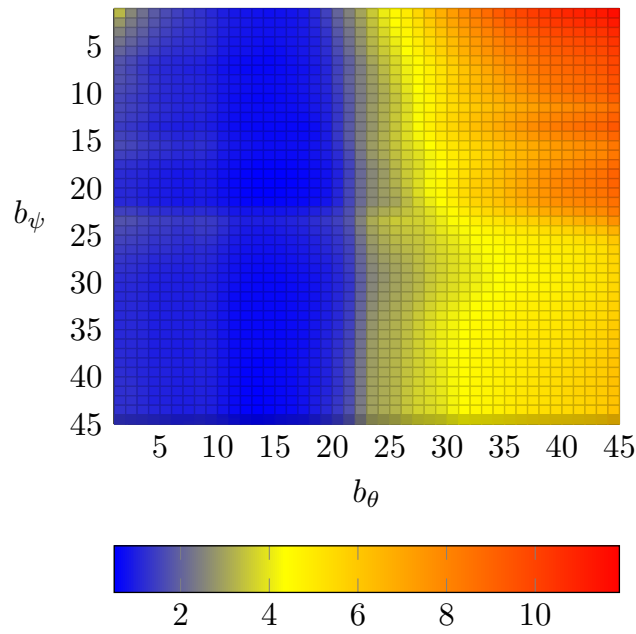
0,7 % und 2,9 % liegen, weichen nur geringfügig von den pro Datensatz optimierten Kosten c_{II} ($0,3 \% \leq c_{II} \leq 2,0 \%$) aus Tabelle 4.9 (Kapitel 4.4.1) ab. Deshalb kann angenommen werden, dass die Detektion aus Kapitel 3.4.4 in Kombination mit der Segmentierung aus Kapitel 3.4.3 (mit dem Parametersatz $\{\alpha, t_b, T_\psi, T_\theta\}$) zwar robust gegenüber Variationen des Parameters b_ψ ist, jedoch wenig robust gegenüber Veränderung des Parameters b_θ .

Abbildung 5.4 zeigt beispielhaft die Kosten c_{II} (nach Gleichung 3.112 aus Kapitel 3.7.5) zur Variation der Parameter b_ψ und b_θ für den ersten Lerndatensatz (siehe Kapitel 3.7.5). In Abbildung 5.4 ist die genannte Abhängigkeit vom Parameter b_θ erkennbar. Die Bereiche, in denen die Kosten c_{II} unterhalb von 0,6 % liegen, befinden sich mit wenigen Ausnahmen in dem Bereich, in dem der Parameter b_θ zwischen 13° und 15° liegt. Wohingegen innerhalb dieses Bereiches ($13^\circ \leq b_\theta \leq 15^\circ$) der Parameter b_ψ nur wenig Einfluss hat. So liegen die Kosten c_{II} in dem Bereich $13^\circ \leq b_\theta \leq 15^\circ$ ohne eine Beschränkung durch b_ψ bei maximal 1,02 %.

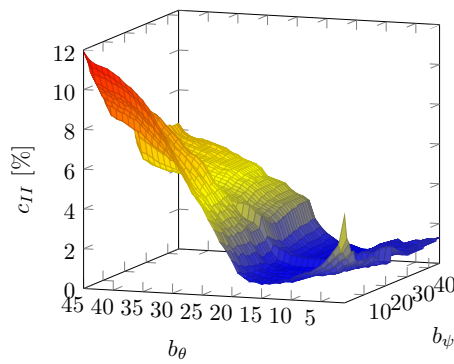
Einige Ergebnisse der Tabelle 4.9 aus Kapitel 4.4.1, welche die nach der Kostenfunktion aus Gleichung 3.112 (Kapitel 3.7.5) je Datensatz optimierten Ergebnisse führt, sind größer als die der Tabellen E.29-E.31 zur *Parameterwahl II* aus Anhang E.2.2. Die Diskussion, den Generalized Pattern Search (GPS) Algorithmus [Kolda u. a. 2003] zur *Parameteroptimierung II* in Kapitel 3.7.5 einzusetzen, lässt sich wie im vorherigen Kapitel (5.3.1) führen. Jedoch wäre es in der selben Zeit mit dem Verfahren der simulierten Abkühlung (SA - *simulated annealing*) [Buseti 2003] nicht möglich gewesen, eine entsprechende Vielzahl an Datensätzen (15 Datensätze mit je 199 Bildern) und Parameterkombinationen (6 unterschiedliche, siehe Kapitel 3.7.5) zu untersuchen. Die Trennung der Optimierung in die *Parameteroptimierung I* und *Parameteroptimierung II* aus Kapitel 3.7.5 hat zudem dazu beigetragen sowohl mehr Details zu den Einzelschritten zu erfahren, als auch die Parameteroptimierung zu beschleunigen. Eine Parameteroptimierung aller Parameter in Kombination der einzelnen Parametersätze wurde nicht durchgeführt, da die Kostenfunktionen darauf ausgelegt sind entsprechend relevante Informationen zu den Teilaufgaben der Algorithmen zur Detektion aus Kapitel 3.4 zu optimieren.

Die durchschnittliche Genauigkeit (AP - *average precision*, siehe [Zhu 2004; Murphy 2012]) der Detektionsergebnisse aus Kapitel 4.4.2 liegt für die untersuchten Testdatensätze zwischen 87,7 % und 95,2 % und für die 15 Testdatensätze im Mittel bei 93,2 %. Diese liegen deutlich oberhalb der in [Zhang u. a. 2013] beschriebenen Detektionsraten (maximal 58,4 % für Flugzeuge), wobei in dieser Dissertation lediglich Pferdeköpfe vom restlichen Bildinhalt zu trennen sind und nicht wie in [Zhang u. a. 2013] zwischen mehreren Klassen unterschieden wird. Mit den in Tabelle 4.10 aus Kapitel 4.4.2 angegebenen Treffergenauigkeiten (ACC) von 96,5 % bis 98,4 %, welche unter realen Bedingungen erzielt wurden, ist ein praktischer Einsatz denkbar. Das Verfahren aus Kapitel 3.4.3, welches zur Detektion eingesetzt wurde, kommt ohne das Anlernen des Hintergrundes aus (vgl. Kapitel 5.3.1). Dadurch lässt es sich auch ohne eine fest installierte Tiefenkamera einsetzen. Die in dieser Dissertation

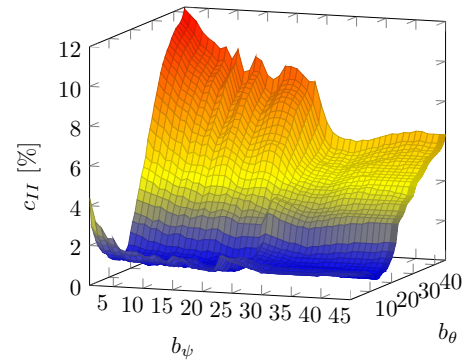
5. Diskussion und Fazit



(a) Draufsicht



(b) Seite 1



(c) Seite 2

Abbildung 5.4.: Kosten zur Parametervariation der Parameter b_ψ und b_θ des ersten Lerndatensatzes

erzielten Detektionsergebnisse aus Kapitel 4.4.2 wurden jedoch ohne eine Veränderung des Hintergrunds erzielt.

Fazit

Zur Optimierung der Parameter der entwickelten Segmentierung und der angepassten Klassifikation zur Detektion der Pferdeköpfe wurden Kostenfunktionen erstellt, so dass aus 15 Lerndatensätzen zu je 199 Frames die optimalen Parameter für eine bestmögliche Segmentierung bestimmt werden konnten. Mit diesen Parametern wurde eine durchschnittliche Genauigkeit (AP - *average precision*) von bis zu 95,2 % bzw. eine Treffergenauigkeit von 96,5 – 98,4 % auf den 15 Testdatensätzen (zu je 199 Frames) beim Auffinden der Pferdeköpfe in den Bilddaten erzielt.

Zur Erweiterung des beschriebenen Verfahrens zur Klassifikation (Detektion) aus Kapitel 3.4.4 ist der Einsatz der durch Cover u. Hart [1967] oder Bhatia u. a. [2010] beschriebenen Algorithmen der Nächsten-Nachbarn (NN - *nearest-neighbour*) oder k-Nächsten-Nachbarn (KNN - *k-nearest-neighbour*) sinnvoll. In Kapitel 3.4.4 wurde lediglich der Abgleich mit einem Template \mathcal{H}_T durch den mit der Gleichung 3.55 beschriebenen Ähnlichkeitsfaktor ρ eingesetzt. Dadurch ist die Auswertung aus Kapitel 4.4 deutlich vereinfacht, da die Klassifikation nach den k-Nächsten-Nachbarn (KNN) stark von den Nachbarn (Einträge in der Datenbank) abhängt. Mit den Algorithmen der k-Nächsten-Nachbarn (KNN) ließe sich die Detektion jedoch besser auf individuelle Kopfformen (z.B. unterschiedliche Pferderassen oder Tierarten wie Rinder) anpassen, da die Abweichungen zu den Nachbarn deutlich enger gewählt werden können als zu einem einzigen Template. Bei der richtigen Wahl der Nachbarn ließen sich damit vermutlich noch bessere Detektionsergebnisse als in Kapitel 4.4.2 erzielen.

5.4. Posenbestimmung, Landmarkendetektion und Normalisierung

Das Auffinden von Landmarken ist entsprechend der Diskussion zur *Detektion* aus Kapitel 5.3 auf 2D Bilddaten bei Pferden durch die unterschiedlichen Zeichnungen und die Orientierung der Augen zu den Seiten des Kopfes nur schwer möglich. Das Verfahren aus Kapitel 3.5.2 zur *Landmarkendetektion* nutzt daher zur Auswertung die Tiefeninformationen, welche durch das in Kapitel 3.3 beschriebene System aufgezeichnet und bereits zur *Detektion* der Pferdeköpfe (Kapitel 3.4) eingesetzt wurden. Die *Landmarkendetektion* aus Kapitel 3.5.2 verwendet dabei die durch die Posenbestimmung aus Kapitel 3.5.1 bzw. 3.5.2 normalisierten Daten. Der Großteil der in Kapitel 2.3.2 vorgestellten Arbeiten zur Posennormalisierung bei Tieren setzt auf manuell gesetzte Marker anstatt diese, wie in dieser Dissertation beschrieben, aufzufinden. Beim Menschen wird hingegen zur automatischen Landmarkendetektion auf Tiefendaten häufig die Nase als markantes Merkmal im Gesicht genutzt (siehe Kapitel 2.3.1). Die Nüstern der Pferde (als Pendant zur Nase des Menschen) bieten hingegen keine entsprechende Markanz, da diese nicht wie die Nase beim Menschen aus den restlichen Tiefendaten herausragen und dementsprechend nicht so deutlich aus unterschiedlichsten Blickrichtungen auf den Kopf erkennbar sind.

5. Diskussion und Fazit

Die Posennormalisierung aus Kapitel 3.5.3 berücksichtigt zur Transformation des originalen Grauwertbildes in eine Frontansicht des Pferdekopfes die Tiefendaten der Kamera sowie die durch die Verfahren aus Kapitel 3.5.1 und 3.5.2 bestimmte Pose des Kopfes im \mathbb{R}^3 . Die Grauwertbilder der Pferdeköpfe werden so durch eine nicht-lineare Transformation im \mathbb{R}^2 unter Berücksichtigung der tatsächlichen Kopfform der Pferde im \mathbb{R}^3 in die Frontansichten transformiert. In den meisten in der Literatur beschriebenen Arbeiten aus Kapitel 2.3.2 zur Normalisierung der Pose wird vor der Identifikation eine Affine Transformation des zweidimensionalen Bildes eingesetzt. Die Affine Transformation stellt jedoch lediglich eine lineare Transformation in den zweidimensionalen Bilddaten dar. Diese reichen nach Zhang u. Gao [2009] jedoch nicht aus, um die Variationen durch unterschiedliche Posen in den Bilddaten zu korrigieren.

Zur zunächst groben Posenbestimmung in Kapitel 3.5.1 wird die spezielle Kopfform der Pferde ausgewertet. Die Köpfe der Pferde sind in Höhe der Augen (Stirn direkt oberhalb) deutlich breiter als am langen schmalen Nasenrücken, auch die Nüstern und das Maul sind nicht breiter als die oberen Bereiche des Schädels (vgl. [von den Driesch 1976, Seite 20, Abbildung 5a - Stirnbreite Nr. 41]). Dies wird mit der Bestimmung des Masseschwerpunktes \mathbf{M}_C in Kapitel 3.5.1 ausgenutzt, um eine grobe Orientierung der Pferdeköpfe zunächst auf der Maske des detektierten Pferdekopfes und anschließend im Raum zu bestimmen. Auch die in Kapitel 3.5.2 beschriebene *Posenverfeinerung* ist auf die besondere Kopfform ausgelegt. Hier wird ein geometrisches Teilmodell (vgl. Kapitel 2.3.1) bestehend aus mehreren Kreisen in die dreidimensionalen Tiefendaten der Pferdeköpfe eingepasst. Aus den Kreiszentren der eingepassten Kreise im \mathbb{R}^3 kann nach der Beschreibung aus Kapitel 3.5.2 die Medianebene extrahiert werden. Die breiteste Stelle nach der Posennormalisierung am Pferdekopf ist der Bereich, in dem die Landmarken nach Kapitel 3.5.2 extrahiert werden. Die Landmarken liegen damit leicht oberhalb der Augen (vgl. [von den Driesch 1976, Seite 20, Abbildung 5a - Ectorbitale]).

Die Parameter zur Posennormalisierung und Landmarkendetektion der Pferdeköpfe sind nach Kapitel 3.7.6 über die Vergleichsdaten aus Kapitel 3.7.3 optimiert. Die Posen der Pferdeköpfe in den Vergleichsdaten (Kapitel 3.7.3) wurden hierzu zuvor mit dem ICP-Algorithmus (*iterative closest point - algorithm*, siehe Anhang D) bestimmt. In Kapitel 2.3.1 wurde bereits beschrieben, dass der ICP-Algorithmus (*iterative closest point - algorithm*) das Standardverfahren zur Registrierung rigider (nicht verformbarer) Körper ist (siehe [Gökberk u. a. 2009]). Dadurch findet ein Vergleich der Posendetektion in Kombination mit der Landmarkendetektion aus Kapitel 3.5 durch die in Kapitel 3.7.6 beschriebene Kostenfunktion c_A aus Gleichung 3.116 bzw. c_B aus Gleichung 3.122 mit den Resultaten des ICP-Algorithmus statt.

Der ICP-Algorithmus kann die in Kapitel 3.5 beschriebene Posennormalisierung (bzw. Posenbestimmung) nicht ersetzen, da er auf eine gute Initialisierung angewiesen ist (siehe Kapitel 2.3.1 bzw. Anhang D) und unterschiedliche Skalierungen des

einzupassenden Modells unberücksichtigt bleiben. In Kapitel 3.7.3 wurde daher pro Tier ein individuelles Kopfmodell verwendet, um die unterschiedlichen Größen der Pferdeköpfe zu berücksichtigen. Außerdem sind Marker für eine Initialisierung zuvor von Hand gesetzt worden, um die Vergleichsdaten aus Kapitel 3.7.3 zu erstellen. Entsprechende Marker stehen jedoch vor einer automatischen *Landmarkendetektion*, wie sie in Kapitel 3.5.2 beschrieben ist, nicht zur Verfügung.

Die durchschnittlichen Fehler $c_{\emptyset B}$ bzw. $c_{\hat{\emptyset} B}$ pro Frame ($c_{\hat{\emptyset} B} = \frac{c_B}{199}$ mit Kostenfunktion c_B nach Gleichung 3.122 aus Kapitel 3.7.6, $c_{\emptyset B}$ für Lern- und $c_{\hat{\emptyset} B}$ für Testdaten) liegen in der Tabelle 4.18 aus Kapitel 4.5.1 (pro Datensatz optimiert nach c_B) zwischen 3,3 und 4,6 cm und in den Tabellen F.1-F.5 aus Anhang F (einheitliche Parameter für alle Datensätze) zwischen 3,5 und 3,8 cm. Der Augendurchmesser der Pferde liegt nach Howland u. a. [2004] bei ca. 4 cm (Abhängig von der Rasse, Holsteiner nicht aufgeführt), so dass der bei der Posenkorrektur feststellbare Fehler (nach $c_{\emptyset B}$ bzw. $c_{\hat{\emptyset} B}$) in Bezug auf die Ergebnisse des ICP-Algorithmus (*iterative closest point - algorithm*) in etwa einer Augenbreite entspricht.

Die Optimierung der Parameter zur Posenbestimmung durch den Generalized Pattern Search (GPS) Algorithmus nach Kapitel 3.7.6 lässt sich wie zuvor zur *Detektion* (siehe Kapitel 5.3.1 bzw. 5.3.2) diskutieren. Jedoch sind vergleichbare Darstellungen des Parameterraumes im Bereich der *Parameterwahl* nach Kapitel 4.5.2, wie sie in Kapitel 5.3 gezeigt wurden, durch die hohe Anzahl an Parametern (7 Parameter zur *Parameteroptimierung* B aus Kapitel 3.7.6) deutlich unübersichtlicher. Durch die eigenen Untersuchungen wurde jedoch festgestellt, dass mit dem Parameter T_N bei 100 % der Parameter T_O keinen Einfluss auf die Ergebnisse der Kostenfunktion c_A aus Kapitel 3.7.6 hat.

Die Validierung der Posennormalisierung wird in der Literatur häufig über die Erkennungsraten der Identifikationsalgorithmen für Gesichter mit unterschiedlichem Winkel zur Kamera bestimmt z.B. [Gökberk u. a. 2009; Yi u. a. 2013]. Die Auswertung der Identifikation findet sich im folgenden Kapitel 5.5, wobei bewusst die Eigengesichter aus Kapitel 3.6.2 zur Identifikation eingesetzt wurden. Der Algorithmus der Eigengesichter (Kapitel 3.6.2) ist stark anfällig gegenüber veränderten Bilddaten der Normalisierung aus Kapitel 3.5.3, so dass dieser bei der Unterscheidung nicht frontaler Bilddaten deutlich schlechter abschneidet [Chai u. a. 2005]. Dadurch sind die in Kapitel 4.6 erreichten Ergebnisse der Identifikation eine zusätzliche, indirekte Validierung aller Schritte der Posennormalisierung aus Kapitel 3.5. Eine Auswertung der Identifikation nach den unterschiedlichen Orientierungen der Pferdeköpfe zur Kamera wäre sinnvoll. Die entsprechende Auswertung wurde jedoch aufgrund weiterer Abhängigkeiten (Auflösung, Lernanteil und Anzahl der Eigenwerte) der Identifikation mit den Eigengesichtern nach Kapitel 3.6.2 in den Ergebnissen aus Kapitel 4.6 nicht berücksichtigt.

Für eine Auswertung mit gleichverteilten Daten pro Kopforientierung reduzieren sich die zur Verfügung stehenden realen Daten pro Position auf den Bereich mit der geringsten Anzahl, welche sich durch den gewählten Winkelbereich pro Orientierung

5. Diskussion und Fazit

bestimmt. Die seitlichen Kopfpositionen sind nach Kapitel 4.3 jedoch deutlich seltener vertreten. Um über die Erkennungsraten der Identifikationsalgorithmen eine aussagekräftige Untersuchung (mit ausreichend gleichverteilten Daten) durchführen zu können sollten daher weitere reale Daten gesammelt werden. Ohne die Kooperation der Tiere braucht das Erfassen der entsprechenden Daten jedoch deutlich mehr aufgezeichnete Besuche. Eine Alternative stellt das Erstellen simulierter Daten dar, wodurch sich zur Auswertung der Posennormalisierung deutlich mehr Daten in den benötigten Positionen generieren lassen. In dieser Dissertation wurden hierzu nur reale Daten ausgewertet.

Fazit

Die eigens entwickelte Posendetektion wurde zunächst auf einer Maske aus den Tiefendaten durchgeführt, um die darüber zunächst nur grob bestimmte Pose anschließend mit einem geometrischen Teilmodell zu verfeinern. Auch hier konnte mit einer passend gewählten Kostenfunktion, welche die Abweichung zu den auf den Vergleichsdaten (mittels dem ICP-Algorithmus *iterative closest point-algorithm*) bestimmten Posen berücksichtigt, ein Satz von optimal einzusetzenden Parametern bestimmt werden. Mit den zusätzlich ermittelten Landmarken, die etwas oberhalb der Augen gefunden wurden, lässt sich eine Posennormalisierung durchführen. Diese Posennormalisierung korrigiert die Ansicht der Pferdeköpfe virtuell dahingehend, dass die Köpfe unabhängig von der ursprünglichen Pose direkt in die Kamera schauen, so dass eine Identifikation auf den Bilddaten möglich wird. Die Grenzen zur Orientierung der Pferdeköpfe wurden bereits in Kapitel 5.2 behandelt. Die durchgeführte Posennormalisierung ermöglicht es selbst mit einem globalen Identifikationsverfahren, die im Allgemeinen anfällig gegenüber Posenvariationen sind, gute Identifikationsresultate zu erzielen.

Die im Zuge der *Posenverfeinerung* aus Kapitel 3.5.2 zu bestimmenden Kreise wurden durch die einfachere algebraische Methode nach Kasa [1976] bestimmt. Der durch die algebraische Methode bestimmte Radius der Kreise fällt im Allgemeinen kleiner aus als der durch die geometrischen Methoden [Al-Sharadqah u. Chernov 2009; Chernov u. Lesort 2005] bestimmte Radius. Sollten die geometrischen Methoden eingesetzt werden, kann dieses Einfluss auf die *Posenverfeinerung* und die *Landmarkendetektion* aus Kapitel 3.5.2 haben, wodurch diese eventuell anzupassen sind.

5.5. Identifikation

Aktuelle 2D Methoden der Gesichtserkennung übertreffen nach Scheenstra u. a. [2005] leistungsmäßig immer noch die 3D Methoden, auch wenn erwartet wird, dass die 3D Methoden in Zukunft die 2D Methoden übertreffen werden [Scheenstra u. a. 2005]. Chang u. a. [2003] vergleicht die Methode der Eigengesichter (siehe Kapitel 3.6.2) auf 2D und 3D Gesichtsdaten bzw. deren Fusion beim Menschen. In [Chang u. a. 2003] zeigt der Vergleich von 2D und 3D Methode, dass die 3D Methode der 2D Methode bei ausreichender Tiefenauflösung der 3D Daten überlegen ist. Werden die Ergebnisse aus [Chang u. a. 2003] hingegen in Abhängigkeit der Tiefenauflösung betrachtet, so würden die 2D und die 3D Methoden unter Verwendung der in dieser Dissertation verwendeten Tiefenkamera aus Kapitel 3.3 bei einer Tiefenauflösung von 5 mm mit vergleichbaren Ergebnissen abschneiden. Eine Tiefenauflösung von 5 mm konnte jedoch nach den Ergebnissen der *Tiefenmessung 1* aus Kapitel 4.2 (siehe 50 % IQA in Tabelle 4.2) nur bis etwas über einen Meter bei frontaler Ausrichtung (vgl. Diskussion um Längenfehler aus Kapitel 5.1.2) erreicht werden.

Zudem macht der Einsatz der 2D Methode die Ergebnisse dieser Dissertation vergleichbar mit den Ergebnissen der Arbeiten durch Kim u. a. [2005b]; Cai u. Li [2013] bzw. Corkery u. a. [2007] zur Erkennung von Rindern bzw. Schafen. Nach Gupta u. a. [2007] bzw. Phillips u. a. [2000] ist die Methode der Eigengesichter auf 2D Bildern (siehe Kapitel 3.6.2) das Referenzverfahren mit dem weitere Verfahren verglichen werden.

Im Gegensatz zu den globalen Verfahren (z.B. PCA - Eigengesichter, LDA - Fisherfaces) sind die lokalen Verfahren (z.B. LBP - *local binary pattern*) nach Zhao u. a. [2003] darauf ausgelegt, Beleuchtungsvariationen und mögliche Verschiebungen (z.B. durch Posenänderung oder unterschiedliche Gesichtsausdrücke) in einem gewissen Rahmen zu kompensieren. Die Hauptteile dieser Dissertation beschreiben die Detektion (siehe Kapitel 3.4) und die Posenkorrektur (siehe Kapitel 3.5), welche für eine automatische Identifikation der Individuen anhand der Kopfaufnahmen von Pferden benötigt werden. Um mögliche Schwachstellen der Posenkorrektur besser erkennen zu können, wird, wie bereits in Kapitel 5.4 beschrieben, ein globales Verfahren eingesetzt. Das Verfahren der Eigengesichter aus Kapitel 3.6.2 ist im Gegensatz zu einigen lokalen Verfahren (z.B. LBP - *local binary pattern*) nicht in der Lage Posenvariationen in den Bilddaten zu kompensieren. Chai u. a. [2005] beschreibt, dass bei Abweichungen von der Frontansicht (z.B. durch eine Posenänderung) die Identifikationsresultate für die globalen Verfahren (z.B. Eigengesichter - PCA und Fisherfaces - LDA) drastisch schlechter werden. Mit den Identifikationsresultaten aus Kapitel 4.6.2 wird durch den Einsatz der globalen Methode aus Kapitel 3.6.2 also indirekt auch das Verfahren der Posenkorrektur (vgl. Kapitel 5.4) validiert. Dies ist der Grund, weshalb in dieser Dissertation das Verfahren der Eigengesichter aus Kapitel 3.6.2 eingesetzt wurde und nicht ein lokales Verfahren, wie in Cai u. Li [2013], mit dem vorrangig bei bestehenden Problemen der Posenkorrektur bessere Identifikationsergebnisse erzielt werden könnten.

Die Ergebnisse aus Kapitel 4.6.2 haben für die Identifikation der Pferde mit dem Algorithmus der Eigengesichter (siehe Kapitel 3.6.2) gezeigt, dass die Auflösung im Vergleich zur Anzahl der eingesetzten Eigenwerte oder des Lernanteils einen nur geringen Einfluss auf die Erkennungsleistung hat. Entsprechendes wurde bereits durch Zhao u. a. [2003] für globale Verfahren bei der Gesichtserkennung von menschlichen Gesichtern festgestellt. Speziell mit dem Verfahren der Fisherfaces (LDA, siehe Kapitel 2.4.2), ebenfalls ein globales Verfahren, kann eine Unterscheidung (Verifikation) von menschlichen Gesichtern bereits mit einer Auflösung von 12×10 durchgeführt werden [Zhao u. a. 2003]. In Kapitel 4.6.2 wurde zudem beobachtet, dass trotz einer höheren Auflösung bei ansonsten gleichen Parametern die Treffergenauigkeit des Verfahrens aus Kapitel 3.6.2 zur Identifikation von Pferden zum Teil schlechter ausfällt als bei einer etwas niedrigeren Bildauflösung. Auch dies konnte bereits in [Zhao u. a. 2003] für menschliche Gesichter festgestellt werden, wobei nach Zhao u. a. [2003] eine Erklärung hierzu ein verbessertes Signal-zu-Rausch-Verhältnis bei niedrigerer Auflösung ist.

Verglichen mit den Arbeiten in der Literatur zur Identifikation von landwirtschaftlichen Nutztieren anhand von Gesichtsaufnahmen aus Kapitel 2.4.4 sind die Ergebnisse dieser Dissertation für den praktischen Einsatz deutlich vielversprechender. Zum einen konnten die Bilddaten zur Identifikation in dieser Dissertation automatisch aus realen Aufnahmen erzeugt und extrahiert werden (siehe Kapitel 3.3-3.5). Lediglich eine Auswahl der „guten“ Bilddaten (Gruppierung der Daten), die in Kapitel 3.7.7 beschrieben ist, wurde bisher nicht automatisiert und ist vermutlich zur Auswahl der Lerndaten auch weiterhin sinnvoll, wobei die in Kapitel 4.6.1 beschriebenen Bedingungen den Großteil der „schlechten“ Bilddaten bereits automatisch herausfiltert. Ohne diese Auswahl ist zwar zu vermuten, dass die Ergebnisse der Identifikation schlechter ausfallen, jedoch wäre ein Vergleich mit den Arbeiten aus Kapitel 2.4.4 nicht möglich, da diese ausnahmslos von Hand markierte Bilddaten (und damit optimale Bilddaten) einsetzen.

Zum anderen fallen die Ergebnisse bei der Identifikation der in dieser Dissertation untersuchten Gruppe von 9 Pferden mittels der aufgenommenen Gesichter im Vergleich zu den genannten Arbeiten aus Kapitel 2.4.4 zur Identifikation von Rindern [Cai u. Li 2013] und Schafen [Corkery u. a. 2007] besser aus. Die Verifikation durch Kim u. a. [2005b] ist nicht mit dieser und den restlichen Arbeiten vergleichbar, da bis auf Transformationen oder Änderungen (z.B. zusätzliches Rauschen) auf den Bilddaten zum Test und zur Verifikation die gleichen Ursprungsbilder eingesetzt wurden. Dadurch lassen sich zwar Tendenzen bezüglich der Anfälligkeit der Algorithmen zur Identifikation durch bestimmte Transformationen oder Bildänderungen prognostizieren, jedoch lassen sich keine Erkennungsraten für unterschiedliche Bilddaten der jeweiligen Tiere vorhersagen. Corkery u. a. [2007] berichtet von einer Trefferquote (Richtig-Positiv-Rate) von 96 % bei einem Lernanteil von 57 % (4 Bilder zum Anlernen und 3 Bilder zum Testen), wohingegen in Tabelle 4.28 aus Kapitel 2.4.4 für den schlechtesten Fall (für das Pferd „Delia“) bereits eine Trefferquote von 96,8 % bei

einem Lernanteil von 50 % erzielt werden konnte. Die durchschnittliche Trefferquote bzw. Treffergenauigkeit (ACC) aus Tabelle 4.28 (Kapitel 2.4.4) liegt bei 99,2 % mit einem Lernanteil von 50 %. Die Trefferquote von 96 % wurde in [Corkery u. a. 2007] jedoch mit der maximalen Anzahl der 200 Komponenten aus der unabhängigen Komponentenanalyse (ICA, vgl. [Delac u. a. 2005]) erreicht. In Kapitel 2.4.4 sind dagegen maximal 54 Eigenwerte (von den 900 maximal möglichen Eigenwerten bei 50 % Lernanteil) zur Identifikation durch den Algorithmus aus Kapitel 3.6.2 berücksichtigt worden und in den genannten Beispielen der Tabelle 4.28 (Kapitel 2.4.4) wurden lediglich 36 Eigenwerte eingesetzt. Mit nur 60 Komponenten wird durch Corkery u. a. [2007] eine Trefferquote um die 86 % angegeben. In [Corkery u. a. 2007] wurden dagegen weniger Bilder (absolute Anzahl) zum Anlernen gebraucht, um mehr Tiere (50 Schafe) unterscheiden zu können. Gegenüberzustellen ist jedoch auch der Aufwand um eine entsprechende Anzahl an Bildern zum Anlernen zu sammeln, welcher durch die in dieser Dissertation vorgestellten automatischen Verfahren (mit Ausnahme der Auswahl entsprechend Kapitel 3.7.7) deutlich reduziert ist.

Cai u. Li [2013] erreicht mit einer Abwandlung des lokalen Verfahrens LBP (*local binary pattern* von Ahonen u. a. [2004]) bei der Identifikation von Rindern innerhalb einer Gruppe von 30 Tieren eine Erkennungsrate (entspr. Treffergenauigkeit (ACC)) von 95,3 %. Cai u. Li [2013] nutzt dazu 90 % der 100 Bilder je Tier als Lerndaten. Die durchschnittliche Treffergenauigkeit der Ergebnisse dieser Dissertation in Tabelle 4.26 aus Kapitel 4.6.2 liegt im Vergleich dazu bei 97 %, wohingegen lediglich 12,5 % der Lerndaten (25 Bilder) genutzt wurden.

Allgemein ist zu berücksichtigen, dass mit einer größeren Anzahl an zu unterscheidenden Individuen mit einer geringeren Erkennungsleistung zu rechnen ist. Beim Menschen wurde mit der Verdopplung der Individuen in einer Datenbank ein Rückgang der Erkennungsraten von zwei bis drei Prozentpunkten beobachtet [Zhao u. a. 2003; Kong u. a. 2005]. In [Phillips u. a. 2005] wird der Einfluss den die Anzahl an Lerndaten (Anzahl der Bilder pro Individuum) hat beschrieben, welcher in dieser Dissertation durch die unterschiedlichen Lernanteile ebenfalls berücksichtigt wurde (siehe Kapitel 4.6.2). Außerdem ist die Veränderung über die Zeit zu berücksichtigen, da je mehr Zeit zwischen zwei Aufnahmen vergeht die Ähnlichkeit und damit die Erkennungsraten bei der Identifikation reduziert ist [Abate u. a. 2007]. Eine Lösung ist nach Abate u. a. [2007] das regelmäßige Erfassen von Bildern, um Änderungen über die Zeit auch in die Datenbank aufzunehmen. Dies ist lediglich in bestimmten Einsatzbereichen praktikabel. So wäre ein Aktualisieren der Bilddaten in einer Futterstation problemlos möglich, da die Tiere in regelmäßigen Abständen neu erfasst werden könnten. Bei einer Verifikation, beispielsweise zur Identitätsfeststellung im Handel mit den Tieren, ist ein entsprechendes Vorgehen jedoch kaum praktikabel.

Fazit

Die erfolgreiche Möglichkeit der Identifikation einzelner Individuen wurde an einer Gruppe von neun Pferden mittels des globalen Verfahrens der Eigengesichter gezeigt. So kann festgehalten werden, dass eine Unterscheidung von Individuen einer kleinen Gruppe von Pferden anhand biometrischer Merkmale im Gesicht der Tiere grundsätzlich möglich ist. Der entsprechende Prozess konnte zudem, bis auf eine letzte Auswahl der Daten vor der Identifikation, automatisiert werden. Im Vergleich zu den Identifikationsresultaten der Literatur bei Rindern und Schafen, konnten auf der beschriebenen Gruppe von Pferden, trotz der automatisierten Detektion und Posen-normalisierung, bessere Ergebnisse bei der Unterscheidung der Tiere erzielt werden. Die Identifikation durch das Verfahren der Eigengesichter ist dabei weniger auf eine hohe Auflösung angewiesen, als dass die Anzahl der Stützvektoren (Eigenvektoren), die den Gesichtsraum aufspannen, und die Anzahl der Lerndaten möglichst hoch sein sollten. Die Treffergenauigkeit (ACC) bei der Identifikation der Individuen aus der Gruppe der neun Pferde liegt beim Einsatz von 36 Eigenwerten, einer Bildauflösung von 30×66 und 50 % bzw. 12,5 % der Lerndaten (100 bzw. 25 Bilder) bei 99,2 % bzw. 97 %. Einzelne Pferde sind mit dem untersuchten Verfahren jedoch schlechter von den anderen Tieren der Gruppe zu unterscheiden, welches auf die Ähnlichkeit der grauwertnormalisierten Bilder zwischen den Tieren zurückzuführen ist. Die Treffergenauigkeit (ACC) liegt in dem schlechtesten Fall, für das Pferd „Delia“, beim Einsatz der 36 Eigenwerte, einer Bildauflösung von 30×66 und 50 % bzw. 12,5 % der Lerndaten (100 bzw. 25 Bilder) bei 96,8 % bzw. 91,9 %.

Für weitere Untersuchungen sollte die Gruppe von Pferden (Anzahl der Tiere) vergrößert werden, wobei die Erkennungsraten von den jeweiligen Tieren innerhalb der Gruppe abhängig zu sein scheint. Mit der in dieser Dissertation eingesetzten Identifikation durch das Verfahren der Eigengesichter aus Kapitel 3.6.2 konnte lediglich ein Algorithmus auf einer kleinen Gruppe von Pferden gezeigt werden. Es bleiben jedoch weitere globale Algorithmen, welche besser darauf ausgelegt sind die Unterschiede zwischen den einzelnen Individuen zu berücksichtigen (z.B. ICA oder LDA, siehe [Delac u. a. 2005]), oder lokale Verfahren, wie das LBP-Verfahren (*local binary pattern*) von Ahonen u. a. [2004], zur Gesichtserkennung, die gute Ergebnisse bei der Gesichtserkennung erzielen und besser skalierbar mit der Anzahl der Datenbankgröße sind (inkrementelles Anlernen möglich). López u. Ruiz [2010] vergleicht hierzu mehrere Abwandlungen des LBP-Verfahrens (*local binary pattern*) von Ahonen u. a. [2004]. Letztlich ist die Auswahl des Identifikationsalgorithmus immer anwendungsspezifisch zu treffen. Der Vergleich weiterer Algorithmen zur Identifikation von Pferden bleibt daher ein Thema für zukünftige wissenschaftliche Untersuchungen.

Ähnlich verhält es sich mit der Normalisierung der Grauwerte. Das in dieser Arbeit gewählte Verfahren aus Kapitel 3.6.1 ist beispielsweise nicht in der Lage starke Beleuchtungsvariationen innerhalb eines Bildes zu kompensieren. Vorangig beim Einsatz der globalen Verfahren zur Gesichtserkennung lassen sich bei inhomogener Beleuchtung (z.B. einseitige Beleuchtung) deutlich bessere Ergebnisse mit den Ver-

fahren, die eine Normalisierung auf Basis der diskreten Kosinustransformation (DCT - *discrete cosine transform*) durchführen, erzielen, als mit der in dieser Dissertation eingesetzten Histogrammäqualisation [Goel u. Nehra 2011]. Die lokalen Verfahren wie LBP [Ahonen u. a. 2004] zeigen auch ohne eine Grauwertnormalisierung gute Ergebnisse.

Die Identifikation der Nutztiere anhand der aufgenommenen Gesichter könnte um eine Analyse des Verhaltens, wie beispielsweise in [Morrow-Tesch u. a. 1998] beschrieben, erweitert werden. Diese könnte es ermöglichen Verhaltensmuster zu studieren und in der Praxis Fehlverhalten oder Krankheiten frühzeitig zu erkennen, die mit der heutzutage eingesetzten Identifikation über die RFID-Transponder nicht erkennbar sind (Position und Haltung der Tiere bleibt unbekannt).

Die in dieser Arbeit beschriebenen Ansätze zur Identifikation von Pferden sind, neben den vorgeschlagenen Arbeiten zur Optimierung der Verfahren, auf weiteren Nutztieren zu untersuchen. Die vorgestellten Methoden sind dabei die Brücke zwischen dem Erfolg der rechnergestützten Identifikation von Menschen anhand ihrer Gesichter und dem Bedarf zur Identifikation von Individuen in der Tierhaltung ohne den Einsatz von am Tier angebrachten Transpondern oder Markern.

6. Zusammenfassung

Die Identifikation von Individuen beim Menschen anhand von Gesichtsaufnahmen wurde bereits in zahlreichen Arbeiten behandelt. Mit aktuellen Methoden werden sogar Erkennungsraten erreicht, die den Fähigkeiten des Menschen zur Unterscheidung einzelner Personen gleich kommen. Dagegen umfasst die Literatur nur wenige Arbeiten, welche entsprechende Ansätze der Gesichtserkennung beim Menschen zur Unterscheidung von Nutztieren untersuchen. Zudem konnten keine Arbeiten ausgemacht werden, die eine Detektion der Köpfe von Tieren (Nutztiere wie Rind, Pferd und Schaf) mit seitlich am Kopf ausgerichteten Augen behandeln. Ein entsprechendes System zur Identifikation von Nutztieren anhand ihrer Gesichter per Kamera stellt eine Alternative zur Erkennung durch am Tier befestigten RFID-Transpondern zum Monitoring der Tiere in der Präzisionstierhaltung (PLF - *precision livestock farming*) dar.

Das Ziel der vorliegenden Arbeit war die weitestgehend automatische Identifikation von Pferden (siehe Kapitel 3.1) anhand der Kameradaten in einer Futterstation (siehe Kapitel 3.2) unter Verwendung der biometrischen Merkmale am Kopf der Tiere. Die Hauptaufgaben dieser Arbeit bestand dabei darin die vor der Identifikation durchzuführenden Schritte der Aufnahme, Detektion und Normalisierung zu entwickeln und zu validieren, wodurch eine entsprechend automatisierte Identifikation erst realisierbar wird. Da in der Literatur keine relevanten Arbeiten zur Identifikation von Pferden aus Bilddaten bekannt sind, musste zunächst eine Datenbank aufgebaut werden. Hierzu wurde eigens ein Bilderfassungssystem aufgebaut, welches es erlaubt die Tiefendaten der Szene parallel zu den Grauwertdaten zweier Industriekameras zu erfassen (siehe Kapitel 3.3). In der Zeit vom 13.04.2012 bis zum 07.05.2013 konnte so eine Datenbank aus 1064 Besuchen erstellt werden, die mehr als 587 k Frames (entspricht etwa 32,5 Stunden) mit ca. 89 k detektierten Pferdeköpfen umfasst (siehe Tabelle 4.1 aus Kapitel 4.1 bzw. Tabelle 4.22 aus Kapitel 4.6.1).

Zur Detektion wurden zwei Methoden (siehe Kapitel 3.4) zum Auffinden der Pferdeköpfe in den Tiefendaten erstellt und verglichen. Diese wurden mit unterschiedlichen Parametern auf den Vergleichsdaten (siehe Kapitel 3.7.3) getestet, wobei die bessere Methode eine durchschnittliche Treffergenauigkeit (ACC - *accuracy*) von 97,4 % (für einen der 15 Testdatensätze sogar bis zu 98,4 %) erreicht (siehe Kapitel 4.4). Die auf die Detektion folgende Posenbestimmung (siehe Kapitel 3.5) wird zu Beginn auf der Tiefenmaske durchgeführt, um die Pose der Köpfe zunächst grob zu bestimmen. Mit einem geometrischen Teilmodell, welches eigenständig speziell für die Kopfform der Pferde entwickelt wurde, konnte die Pose verfeinert werden. Mit Hilfe dieser verfeinerten Pose können automatisiert Landmarken knapp oberhalb der Augen der

6. Zusammenfassung

Tiere ausgemacht und damit letztendlich eine Posennormalisierung durchgeführt werden. Die Detektion, die Posenbestimmung und die Landmarkendetektion wurden ausschließlich auf den Tiefendaten durchgeführt, um unabhängig von Fellfarbe und -zeichnung der Tiere diese in den Bilddaten zu erfassen und deren Kopfpose bestimmen zu können. Die Posennormalisierung nutzt sowohl die Pose als auch die Landmarken aus den Tiefendaten, um die Grauwertbilder unter Berücksichtigung der individuellen Kopfform jedes Tieres zu transformieren. Durch die Posennormalisierung wurde eine virtuelle Kameraansicht aus den Grauwertdaten erzeugt, in der die Pferde, trotz abweichender Kopfhaltung, direkt in die Kamera schauen.

Mit den normalisierten Grauwertdaten wurde die Identifikation der Individuen aus einer Gruppe von neun Pferden mittels des globalen Verfahrens der Eigengesichter (siehe Kapitel 3.6) untersucht. Die Methode der Eigengesichter ist ein Standardverfahren zur Gesichtserkennung von menschlichen Gesichtern und wird häufig in der Literatur zum Vergleich herangezogen. Validiert wurde die Identifikation mit unterschiedlicher Auflösung in den Bilddaten, unterschiedlichen Lernmengen und Dimensionen im Merkmalsraum. Mit dem Einsatz von 36 Eigenwerten, 12,5 % der Lerndaten (25 Bilder pro Pferd) und einer Bildauflösung von 30×66 konnte eine Treffergenauigkeit (ACC - *accuracy*) von 97 % bei der Identifikation der Tiere erreicht werden (siehe Kapitel 4.6.2). Es konnte gezeigt werden, dass die Identifikation bereits mit einer recht geringen Auflösung der Bilddaten möglich ist, wobei die Anzahl der genutzten Eigenwerte und vorrangig die Menge der Lerndaten großen Einfluss auf die Erkennungsleistung haben. Es bleibt jedoch zu beachten, dass teilweise einzelne Pferde deutlich schlechter von den restlichen Tieren zu unterscheiden sind (Treffergenauigkeit von 91,9 % mit den genannten Parametern).

Im Rahmen dieser Arbeit konnte eine weitestgehend automatisierte Identifikation von Pferden anhand von realen Bilddaten der Tierköpfe erfolgreich unter den realen Bedingungen in einem Pferdestall gezeigt werden. Vorrangig die vorgestellte Detektion und Posenbestimmung stellen für den beschriebenen Einsatz neue, in der Literatur bisher nicht vorhandene Methoden dar. Die angewendeten, entwickelten und validierten Methoden bilden eine erste Grundlage zur automatisierten Identifikation von Nutztieren mit seitlich am Kopf ausgerichteten Augen auf Bilddaten unter Verwendung der biometrischen Merkmale.

A. Angaben nach dem Pferdepass der Tiere

Tabelle A.1 listet die Geburtsdaten und das Geschlecht der bereits in Kapitel 3.1 vorgestellten Tiere auf. Der vollständige Name der Tiere ist durch die Erweiterung in Klammern angegeben. Ausnahme ist das Tier „Rosi“, das mit vollem Namen „Wilde Rose“ heißt.

Tabelle A.1.: Geburtsdatum und Geschlecht der Tiere

Name	Geburtsdatum	Geschlecht
Anna (Lang)	30.04.2008	weiblich
Arielle	19.05.2008	weiblich
Baldo	1995	männlich
Bara (Barynia)	09.05.2009	weiblich
Blüte	19.05.2009	weiblich
Blütenzauber	20.06.2009	weiblich
Delia	25.05.2011	weiblich
Diadem (56)	1989	weiblich
Dumbledore	20.04.2011	männlich
Linus	März 2011	männlich
Romanze	März 2001	weiblich
Rosi (Wilde Rose)	25.04.2006	weiblich
Royal Bess	2000	weiblich
Serenade	1997	weiblich

Die folgenden Angaben sind lediglich für die neun Pferde gemacht, deren Grauwertbilder zur Auswertung der Identifikation in Kapitel 4.6 genutzt wurden. Die Tabelle A.2 gibt den Ursprung der Tiere an, wobei zunächst das Tier und anschließend Vater sowie Mutter mit Namen genannt sind. In den jeweils darauffolgenden Zeilen sind die Lebensnummern der Tiere selbst bzw. des Vaters und der Mutter angegeben. Aus der Tabelle A.2 wird deutlich, dass „Anna,, und „Blüte“ sowie „Bara “ und „Rosi“ Geschwister sind. Außerdem haben „Blütenzauber“ und „Dumbledore“ beide „Romanze“ zur Mutter.

A. Angaben nach dem Pferdepass der Tiere

Tabelle A.2.: Ursprung der Tiere mit Lebensnummern

Tier	Vater	Mutter
Anna (Lang) DE421000229208	Legolas DE421000278203	Meine Myrthe DE321210020797
Arielle DE421000246108	Con Air DE321210133897	Romanze DE421000191301
Bara (Barynia) DE421000131109	Lordanos DE321210188993	Diadem II DE321210127689
Blüte DE421000132009	Legolas DE421000278203	Meine Myrthe DE321210020797
Blütenzauber DE421000131509	Dolany DE321210037297	Romanze DE421000191301
Delia DE421000134411	Cassilano DE421000185407	Wilde Rose DE421000146106
Dumbledore DE421000133711	De Chirico DE421000138704	Romanze DE421000191301
Romanze DE421000191301	Diadem (56) -	Dream of Glory -
Rosi (Wilde Rose) DE421000146106	Lordanos DE321210188993	Diadem II DE321210127689

In Tabelle A.3 sind die zum Kopf im Pferdepass geführten “Abzeichen bei Fuß der Mutter” zusammengefasst. Diese unterscheiden sich zum Teil von den Zeichnungen zum Zeitpunkt der Aufnahmen, die in Tabelle 3.1 aus Kapitel 3.1 angegeben sind.

Tabelle A.3.: Vermerk zum Kopf unter „Abzeichen bei Fuß der Mutter“

Name	Vermerk
Anna (Lang)	Weiss. Blesse, Weiss. Unterlippenfleck
Arielle	Weiss. Stern
Bara (Barynia)	Weiss. Blesse
Blüte	Weiss. Stern
Blütenzauber	Weiss. Blesse, Weiss. Schnippe
Delia	Weiss. Blesse
Dumbledore	Weiss. Stichelhaar
Romanze	-
Rosi (Wilde Rose)	Weiss. durchgehende Blesse

In den Abbildungen A.1 - A.3 ist der für die Zeichnungen am Kopf der Tiere relevante Teil der Diagramme aus den Pferdepässen abgebildet. Die vollständigen Diagramme sind aufgrund der Übersichtlichkeit nicht dargestellt. Das Diagramm im Pferdepass der Tiere stellt eine abstrakte Skizze eines Pferdes dar, in welche alle relevanten Zeichnungen und Merkmale des jeweiligen Pferdes einzuzichnen sind. Die weißen Stellen am Tier sollen dabei im Diagramm lediglich in Rot umrandet werden. Eine Besonderheit stellen die fleischfarbenen Abzeichen dar, die, wie bei „Dumbledore“ und „Rosi“ aus Abbildung A.3a bzw. c, in Rot auszumalen sind. Alle weiteren, nicht weißen Kennzeichen, wie z.B. Brandzeichen oder Narben, müssen im Diagramm in schwarz eingezeichnet sein. Wirbel werden mit einem schwarzen Kreuz markiert.

Abbildung A.1 zeigt einen Teil des Diagramms der Tiere „Anna“ und „Arielle“.

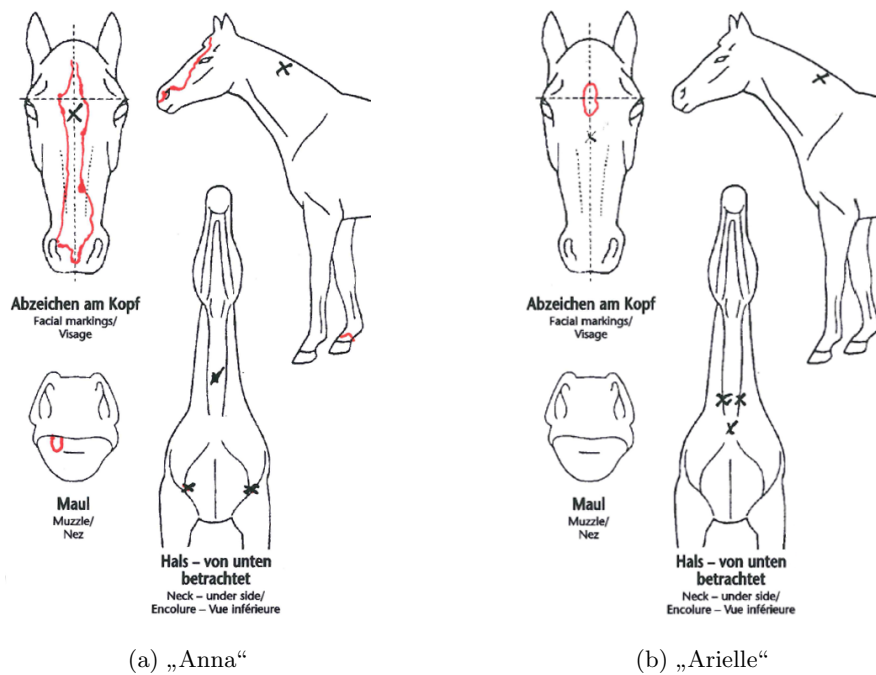


Abbildung A.1.: Diagramme aus den Pässen der Tiere (1)

A. Angaben nach dem Pferdepass der Tiere

Die Abbildung A.2 fasst die Diagramme der Tiere „Bara“, „Blüte“, „Blütenzauber“ und „Delia“ zusammen.

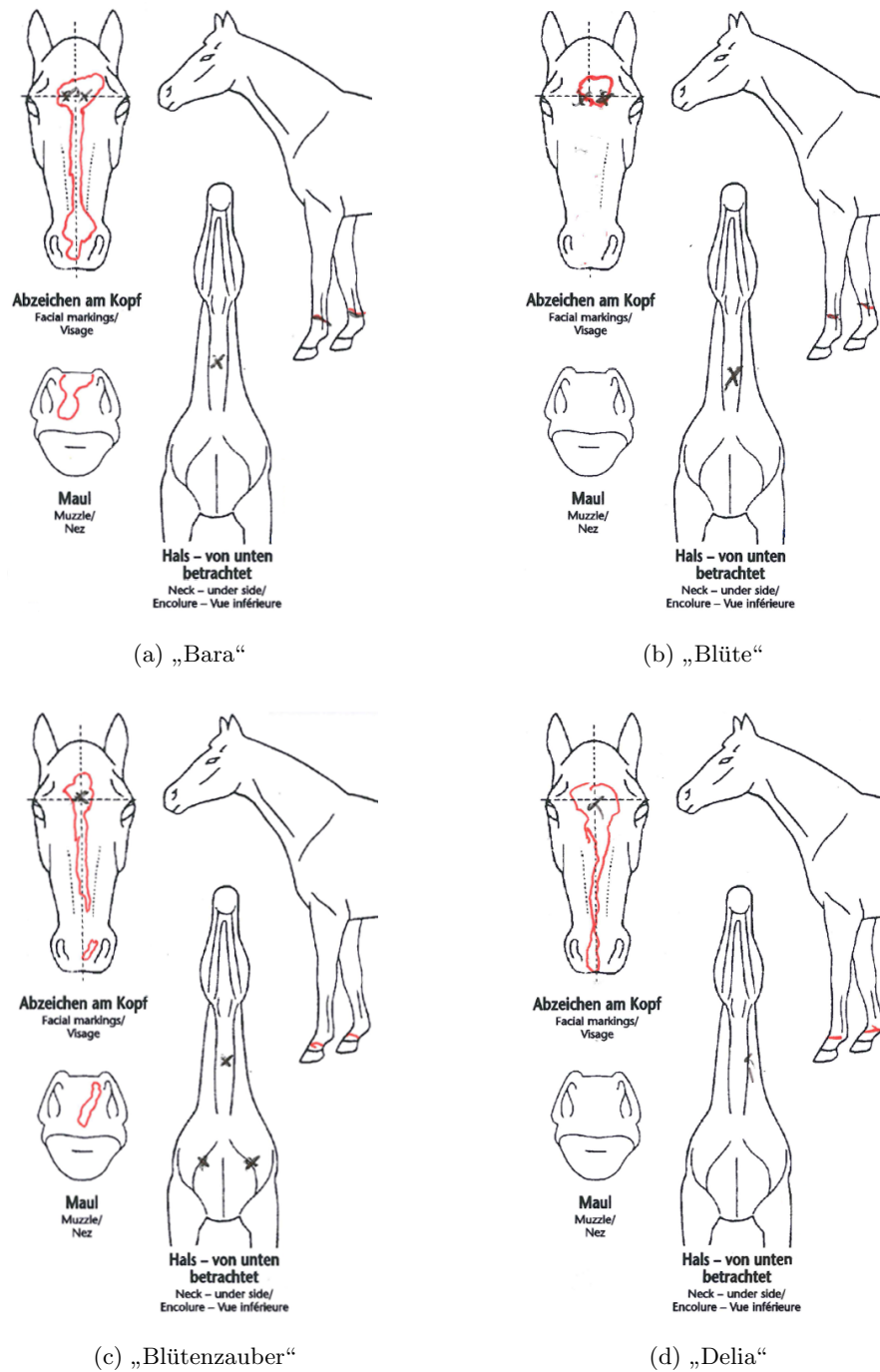
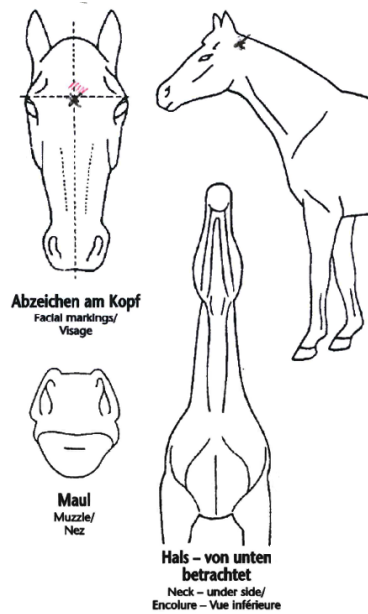
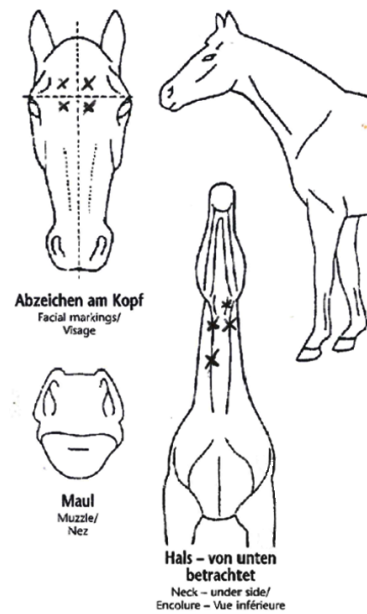


Abbildung A.2.: Diagramme aus den Pässen der Tiere (2)

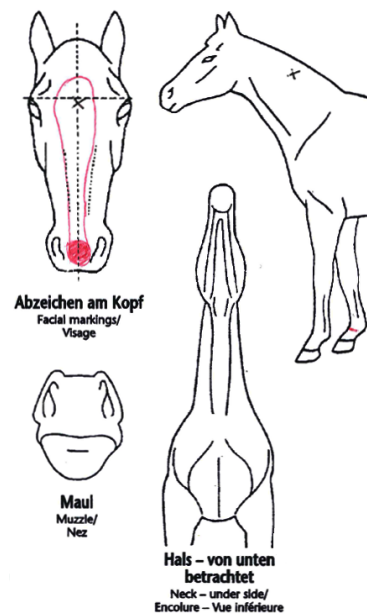
In Abbildung A.3 sind die Diagramme der Pferde „Dumbledore“, „Romanze“, und „Rosi“, dargestellt.



(a) „Dumbledore“



(b) „Romanze“



(c) „Rosi“

Abbildung A.3.: Diagramme aus den Pässen der Tiere (3)

B. Projektive Geometrie

Dieses Kapitel ist eine Kurzfassung der Informationen aus Stahl [2007] und Hartley u. Zisserman [2004], auf die der Leser für weitere Details verwiesen ist. Die Projektive Geometrie erlaubt neben Skalierung, Rotation und Scherung auch eine Translation oder Projektion ($\mathbb{P}^3 \Rightarrow \mathbb{P}^2$) durch eine einfache Matrixmultiplikation zu beschreiben. Ein Punkt in dem projektiven Raum \mathbb{P}^n wird hierzu durch $n + 1$ Koordinaten beschrieben. Zudem wird eine Äquivalenzrelation verwendet

$$\hat{\mathbf{x}} \sim \hat{\mathbf{w}} : \iff \exists \lambda \in \mathbb{R}, \lambda \neq 0, \text{ mit } \hat{\mathbf{x}} = \lambda \hat{\mathbf{w}}, \quad (\text{B.1})$$

wobei $\hat{\mathbf{x}} = [x_1 \ x_2 \ \dots \ x_{n+1}]^T$ und $\hat{\mathbf{w}} = [w_1 \ w_2 \ \dots \ w_{n+1}]^T$ ist. Ein Punkt im Endlichen hat dabei die Koordinate $x_{n+1} \neq 0$ bzw. $w_{n+1} \neq 0$. Punkte im Unendlichen, die zur Beschreibung von Fluchtpunkten verwendet werden können, haben die Koordinate $x_{n+1} = 0$ bzw. $w_{n+1} = 0$. Der Punkt $\hat{\mathbf{x}} = [x_1 \ x_2 \ x_3]^T \in \mathbb{P}^2$ (mit $x_3 \neq 0$) entspricht dabei dem euklidischen Punkt $\mathbf{x} = [\frac{x_1}{x_3} \ \frac{x_2}{x_3}]^T \in \mathbb{R}^2$. In Abbildung B.1 ist ein Beispiel des Punktes $\mathbf{x} = [-2 \ 2]^T \in \mathbb{R}^2$ abgebildet, dieser liegt mit projektiven Koordinaten beschrieben $\hat{\mathbf{x}} = [-2 \ 2 \ 1]^T \in \mathbb{P}^2$ auf der projektiven Ebene ($z = 1$). Durch die Äquivalenzrelation gilt $\hat{\mathbf{x}} \sim \hat{\mathbf{w}} = [-4 \ 4 \ 2]^T \in \mathbb{P}^2$.

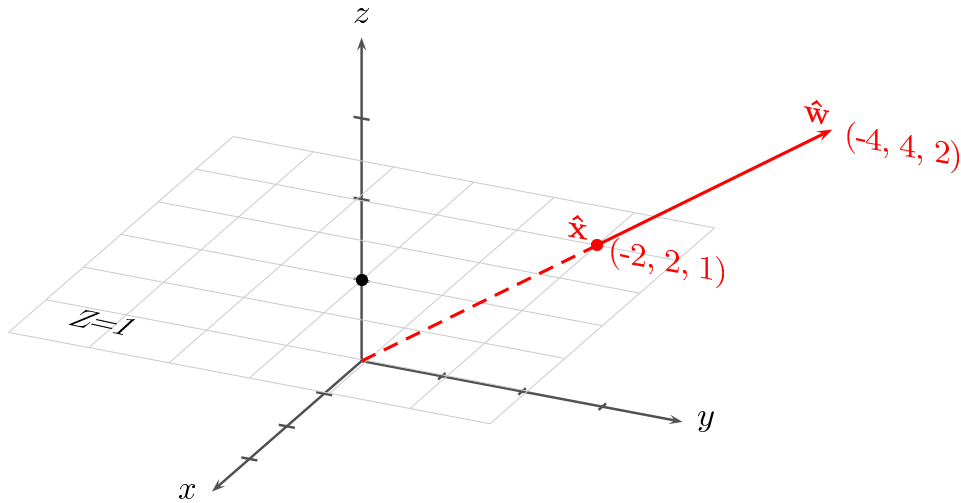


Abbildung B.1.: Projektive Ebene \mathbb{P}^2 ; Euklidische Ebene bei $z = 1$ [Stahl 2007]

Die Projektive Transformation \mathbf{H} wird auch Homografie genannt, sie bildet Punkte aus dem \mathbb{P}^n in sich selber ab ($\mathbb{P}^n \rightarrow \mathbb{P}^n$). Die Transformation muss invertierbar

B. Projektive Geometrie

sein, daher darf die Matrix \mathbf{H} , mit der die Transformation beschrieben wird, nicht singulär sein. Durch die Skalierungsinvarianz, die durch die Äquivalenzrelation zustande kommt, hat \mathbf{H} somit $(n+1)^2 - 1$ Freiheitsgrade. Im \mathbb{P}^2 hat die Projektive Transformation die Form

$$\begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \text{ und im } \mathbb{P}^3 \text{ die Form } \begin{pmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \\ h_{41} & h_{42} & h_{43} & h_{44} \end{pmatrix}.$$

Die Affine Transformation besitzt $n(n+1)$ Freiheitsgrade und hat im \mathbb{P}^2 die Form

$$\begin{pmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{pmatrix}, \text{ im } \mathbb{P}^3 \text{ wird sie durch die Matrix } \begin{pmatrix} a_{11} & a_{12} & a_{13} & t_x \\ a_{21} & a_{22} & a_{23} & t_y \\ a_{31} & a_{32} & a_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \text{ dargestellt.}$$

t_x, t_y und t_z sind dabei Translationen auf den jeweiligen Achsen. Die oberen linken Untermatrizen (alle a Elemente) dürfen nicht singulär sein.

Die Ähnlichkeitstransformation setzt sich zusammen aus einer Skalierung s , einer Rotation (alle r Elemente) und einer Translation (alle t Elemente). Dies ergibt die Form

$$\begin{pmatrix} s r_{11} & s r_{12} & t_x \\ s r_{21} & s r_{22} & t_y \\ 0 & 0 & 1 \end{pmatrix} \text{ für den } \mathbb{P}^2 \text{ und } \begin{pmatrix} s r_{11} & s r_{12} & s r_{13} & t_x \\ s r_{21} & s r_{22} & s r_{23} & t_y \\ s r_{31} & s r_{32} & s r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \text{ für den } \mathbb{P}^3.$$

Die Anzahl der Freiheitsgrade der Rotationsmatrix ergibt sich aus den Kombinationsmöglichkeiten der n Dimensionen $\binom{n!}{(n-2)! 2!} = \frac{n!}{(n-2)! 2}$. Dazu kommen noch $n+1$ Freiheitsgrade für die Translation (n Freiheitsgrade) und die Skalierung (1 Freiheitsgrad). Für den \mathbb{P}^2 hat die Ähnlichkeitstransformation somit 4 Freiheitsgrade (1 Rotation, 2 Translation und 1 Skalierung), im \mathbb{P}^3 sind es 7 Freiheitsgrade (3 Rotation, 3 Translation und 1 Skalierung).

Bei der Euklidischen Transformation fällt die Skalierung weg, so dass diese im \mathbb{P}^2 3 Freiheitsgrade und im \mathbb{P}^3 6 Freiheitsgrade besitzt. Die Matrizen entsprechen

$$\begin{pmatrix} r_{11} & r_{12} & t_x \\ r_{21} & r_{22} & t_y \\ 0 & 0 & 1 \end{pmatrix} \text{ im } \mathbb{P}^2 \text{ und } \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \text{ im } \mathbb{P}^3.$$

Ohne die Rotation bleibt lediglich die Translation, welche n Freiheitsgrade besitzt.

Im \mathbb{P}^2 erfolgt die Translation durch eine Multiplikation mit der Matrix

$$\begin{pmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{pmatrix}, \text{ im } \mathbb{P}^3 \text{ durch die Matrix } \begin{pmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

In der Tabelle B.1 finden sich die Invarianten der jeweiligen Transformation im \mathbb{P}^2 bzw. \mathbb{P}^3 . Die dort aufgeführten Eigenschaften bleiben durch die Transformation erhalten. Da die Gruppe der jeweiligen Zeile eine Teilgruppe der vorangegangenen ist, gelten die aufgeführten Invarianten der jeweiligen Zeile auch für die auf sie folgenden Zeilen.

Tabelle B.1.: Transformationen im \mathbb{P}^2 und \mathbb{P}^3 [Stahl 2007]

Gruppe	Invarianten im \mathbb{P}^2	Invarianten im \mathbb{P}^3
Projektive Transformation	Doppelverhältnis, Linien, Tangenten	-
Affine Transformation	Parallelen, Linie im Unendlichen	Parallele Ebenen, Ebene im Unendlichen
Ähnlichkeitstransformation	Längenverhältnisse, Winkel	Relative Volumen
Euklidische Transformation	Längen, Flächen	Volumen, Flächen, Winkel
Translation	-	-

Wiedererkennbar sind hier die Kameramatrizen aus Kapitel 3.3.2. Die dort beschriebene intrinsische Kameramatrix \mathbf{K} (siehe Gleichung 3.5) ist eine affine Transformation im \mathbb{P}^2 , der Parameter a_{21} ist in dieser speziellen Form immer 0. Die Matrix

$$\begin{bmatrix} \mathbf{R} & -\mathbf{RC} \\ 0 & 1 \end{bmatrix}$$

aus Gleichung 3.7, welche die Orientierung der Kamera mit den extrinsischen Kameraparametern beschreibt, ist eine Euklidische Transformation im \mathbb{P}^3 . Sie setzt sich aus einer Rotation und einer Translation zusammensetzt.

B. Projektive Geometrie

Die Rotationsmatrix \mathbf{R} setzt sich dabei aus den einzelnen Rotationen um die Achsen X, Y und Z mit den Winkeln ψ, θ und ϕ zusammen:

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} = \mathbf{R}_X \mathbf{R}_Y \mathbf{R}_Z \quad \text{mit} \quad \mathbf{R}_X = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\psi) & -\sin(\psi) \\ 0 & \sin(\psi) & \cos(\psi) \end{pmatrix},$$
$$\mathbf{R}_Y = \begin{pmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{pmatrix} \quad \text{und} \quad \mathbf{R}_Z = \begin{pmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Die Projektionsmatrix aus Gleichung 3.3

$$\mathbf{P}_0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

projiziert die Punkte vom \mathbb{P}^3 in den \mathbb{P}^2 .

C. Rekonstruktion der Raumkoordinaten aus den Disparitätenwerten

Die Rekonstruktion beschreibt die Berechnung der Raumkoordinaten aus den Punktkorrespondenzen bzw. deren Disparitätenwerten. Damit ist die Rekonstruktion die entsprechende Umkehr der Projektion aus Kapitel 3.3.2. Die Rekonstruktion aus zwei Kameraperspektiven wird auch als Triangulation bezeichnet. Wie in Abbildung C.1 zu erkennen, ist eine Rekonstruktion des Punktes \mathbf{X} im Raum aus nur einer Kameraperspektive ohne Vorwissen über das Objekt nicht möglich. Bei der Umkehr der Projektion des Pixels \mathbf{x} auf der linken Kameraebene (blau) aus Abbildung C.1 entsteht eine Linie in den Raum hinein, da die Tiefeninformation bei der Projektion verloren geht. Um die Tiefeninformation rekonstruieren zu können ist eine zweite Ansicht des Objekts bzw. des Punktes \mathbf{X} im Raum nötig.

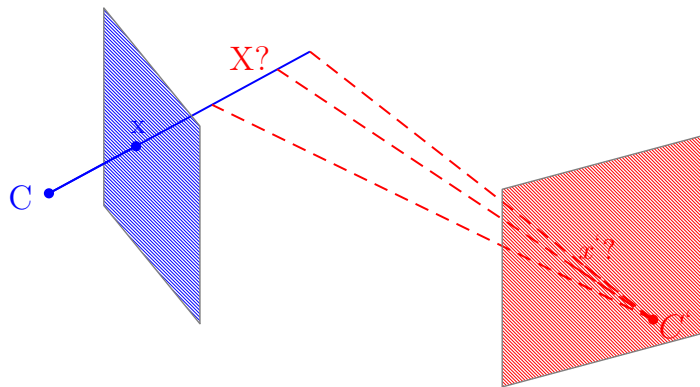


Abbildung C.1.: Tiefeninformation durch Stereo-Vision

Die Ansicht des Punktes \mathbf{X} auf der zweiten Kameraebene (rot, hier rechts) bzw. aus einer zweiten Kameraposition entspricht dem Punkt \mathbf{x}' (rot). Die Punkte \mathbf{x} und \mathbf{x}' werden als Punktkorrespondenzen bezeichnet. Die Disparität beschreibt den Versatz zwischen diesen Punktkorrespondenzen auf einer gemeinsamen, meist virtuellen Kameraebene. Mit der Disparität lässt sich somit, bei bekanntem Punkt \mathbf{x} , der zweite Punkt \mathbf{x}' der Punktkorrespondenz bestimmen. Zusätzlich zu den Punktkorrespondenz werden die intrinsischen und extrinsischen Parameter der Kameras (bzw. deren zwei Ansichten bei nur einer Kamera) zur Rekonstruktion des Punktes \mathbf{X} im Raum benötigt. Diese lassen sich durch eine Kalibrierung, welche in Kapitel 3.3.2 beschrieben ist, offline (also vor einer Rekonstruktion) bestimmen.

In dieser Arbeit wird die in Kapitel 3.3 beschriebene Tiefenkamera Kinect eingesetzt. Diese verwendet ein Projektionsverfahren zur Bestimmung des Disparitätenbilds. Dies bedeutet, dass anstelle der ersten Kamera (von zwei Kameras) ein Projektor eingesetzt wird, welcher Lichtstrahlen in den Raum sendet. Diese treffen auf die Oberfläche der aufzunehmenden Szene und werden von einer Kamera aufgenommen. Der Vorteil liegt darin, dass aktiv ein Muster auf die aufzunehmenden Objekte projiziert wird, welches ein besseres Auffinden von Punktkorrespondenzen ermöglicht. Ohne eine entsprechende Zuordnung der Punktkorrespondenzen, welches nur über entsprechend wiedererkannte Texturen möglich ist, kann die Tiefeninformation nicht rekonstruiert werden. Zudem liefert die beschriebene Tiefenkamera Kinect bereits das Disparitätenbild, eine entsprechende Zuordnung der Punktkorrespondenzen findet bereits innerhalb der Kamera statt.

C.1. Umkehr der Projektion

Die Projektion ist im Kapitel 3.3.2 beschrieben. Die allgemeine Projektion ist hier durch Gleichung 3.8 nach dem *Kameramodell* durch

$$\tilde{\mathbf{x}}_{\text{cam}} = \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \sim \begin{pmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{bmatrix} \mathbf{R} & -\mathbf{RC} \\ 0 & 1 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (\text{C.1})$$

gegebenen. α_x , α_y , s , x_0 und y_0 sind die intrinsischen Parameter der Kamera, welche die Abbildung im Kamerakoordinatensystem beschreiben. \mathbf{R} und $\mathbf{t} = -\mathbf{RC}$ sind die extrinsischen Parameter der Kamera, welche die Orientierung der Kamera im Raum beschreiben.

Die durch Herrera u. a. [2011] im Kapitel 3.3.2 vorgestellte *Kamerakalibrierung* nimmt für die Tiefenkamera keine *Verzerrung* an. Zudem wird die Tiefenkamera durch die *Kamerakalibrierung* so positioniert, dass diese im Ursprung liegt ($\mathbf{C} = [0 \ 0 \ 0]^T$) und die Orientierung der Kamera mit dem des Weltkoordinatensystems übereinstimmt $\mathbf{R} = \mathbf{I}_{3 \times 3}$ (Einheitsmatrix). Somit sind die extrinsischen Parameter für die Rekonstruktion eines Punktes \mathbf{X} im Raum nach der in dieser Arbeit beschriebenen *Kamerakalibrierung* irrelevant. Der Punkt $\mathbf{X} = [X \ Y \ Z]^T$ stimmt mit dem Punkt im Kamerakoordinatensystem $\mathbf{X}_{\text{cam}} = [X_{\text{cam}} \ Y_{\text{cam}} \ Z_{\text{cam}}]^T$ überein $\mathbf{X} = \mathbf{X}_{\text{cam}}$.

Die Gleichung C.1 lässt sich so auf die Gleichung 3.5 der Projektion im Kamerakoordinatensystem (Unterkapitel *Kameramodell* in Kapitel 3.3.2) reduzieren:

$$\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} \sim \mathbf{K} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} X_{cam} \\ Y_{cam} \\ Z_{cam} \\ 1 \end{pmatrix}, \text{ mit } \mathbf{K} = \begin{pmatrix} \alpha_x & s & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (\text{C.2})$$

Mit der weiteren Annahme [Herrera u. a. 2011], dass in dem Modell der Tiefenkamera keine Scherung $s = 0$ auftritt, ergibt sich die interne Kameramatrix \mathbf{K} als

$$\mathbf{K} = \begin{pmatrix} \alpha_x & 0 & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{pmatrix}. \quad (\text{C.3})$$

Die Projektion aus Gleichung C.2 entspricht mit den zuvor beschriebenen Annahmen der Abbildung $\mathcal{P} : (X, Y, Z) \in \mathbb{R}^3 \mapsto \mathcal{P}(X, Y, Z) := (u, v) \in \mathbb{R}^2$. Aufgelöst ergibt sich ohne Scherung $s = 0$ und mit $\mathbf{X} = \mathbf{X}_{cam}$

$$u = \frac{\alpha_x X + x_0 Z}{Z} = \alpha_x \frac{X}{Z} + x_0 \quad \text{und} \quad v = \frac{\alpha_y Y + y_0 Z}{Z} = \alpha_y \frac{Y}{Z} + y_0. \quad (\text{C.4})$$

Zudem ist im Unterkapitel *Kamerakalibrierung* durch die Gleichung 3.17 die Modellierung der Tiefenkamera angegeben:

$$Z_{cam}^d = \frac{1}{\beta(d - \gamma)}. \quad (\text{C.5})$$

Welche die Abbildung $\mathcal{F}_Z : d \in \mathbb{Z} \mapsto \mathcal{F}_Z(d) := Z_{cam}^d \in \mathbb{R}$ der Disparität d auf den Tiefenwert Z_{cam}^d (hier $Z_{cam}^d = Z$, da $\mathbf{X}_{cam} = \mathbf{X}$) beschreibt. β und γ sind die Parameter der Tiefenkamera, welche ebenfalls durch die *Kamerakalibrierung* bestimmt werden.

Die Umkehr der Projektion \mathcal{P} aus Gleichung C.4 entspricht der Abbildung $\mathcal{P}^{-1} : (u, v) \in \mathbb{R}^2 \mapsto \mathcal{P}^{-1}(u, v) := (X, Y, Z) \in \mathbb{R}^3$, welche nicht bijektiv ist und entsprechend der Erklärung anhand von Abbildung C.1 keine Rekonstruktion von \mathbf{X} erlaubt. Erst durch die Rekonstruktion der Tiefeninformation Z durch die Disparität d nach Gleichung C.5 lässt sich die Projektion umkehren $\mathcal{R} : (u, v, Z) \in \mathbb{R}^3 \mapsto \mathcal{R}(u, v, Z) := (X, Y, Z) \in \mathbb{R}^3$ oder $\mathcal{R}_D : (u, v, d) \in \mathbb{R}^3 \mapsto \mathcal{R}_D(u, v, d) := \mathcal{R}(u, v, \mathcal{F}_Z(d)) = (X, Y, Z) \in \mathbb{R}^3$. Die umgekehrte Projektion \mathcal{R} wird dann durch die Gleichung

$$X = \frac{u - x_0}{\alpha_x} Z \quad \text{und} \quad Y = \frac{v - y_0}{\alpha_y} Z \quad \text{mit} \quad Z = \frac{1}{\beta(d - \gamma)} \quad (\text{C.6})$$

beschrieben.

D. Registrieren von 3D Daten durch den Iterative Closest Point - Algorithmus

Das Registrieren von Daten beschreibt die Problematik eine Datenmenge mit Hilfe einer geeigneten Transformation auf eine andere abzubilden. Ziel der Registrierung ist, die beiden Datenmengen durch die Transformation in eine bestmögliche Übereinstimmung zu bringen. Der Iterative Closest Point (ICP)-Algorithmus ist hier ein weitverbreitetes Verfahren, welches mittlerweile in unterschiedlichsten Variationen vom Original [Besl u. McKay 1992] Verwendung findet. Der ICP-Algorithmus zielt darauf ab, ein starres 3D Modell M (hier die 3D Punkte eines zuvor segmentierten Kopfes) in Übereinstimmung mit einer 3D Punktwolke P (die 3D Punkte des aktuell zu untersuchenden Kopfes) zu bringen. Dabei wird versucht eine Transformation zu finden, welche nach der Methode der kleinsten Fehlerquadrate einen möglichst kleinen Fehler zwischen den 3D Punkten P und dem Modell M erzeugt. Der Algorithmus ist, wie der Name schon sagt, iterativ. Entgegen den globalen Verfahren, welche im Allgemeinen deutlich rechenaufwändiger sind, garantiert der Iterative Closest Point - Algorithmus nicht, das globale Minimum zu finden. Das Finden des globalen Minimums hängt stark von der initialen Positionierung der Daten zueinander ab. Bei einer schlechten Ausrichtung der Daten braucht das Verfahren meist deutlich mehr Iterationen und landet nicht selten in einem lokalen Minimum. Zudem ist ein Bestandteil des Algorithmus das Auffinden von Korrespondenzen, welches, sollten die Datenmengen keine Ähnlichkeit haben, sehr aufwändig und instabil ist.

Das Registrieren von Daten, die keine Ähnlichkeit besitzen, macht daher nur bedingt Sinn, da diese nicht zuverlässig aufeinander abzubilden sind. Über die Registrierung kann zwar die Ähnlichkeit zweier Objekte geprüft werden, da der zu optimierende Fehler ein entsprechendes Maß für die Deckung zweier Objekte (Punktmengen) ist. Jedoch ist die Transformation variabel, wodurch die bestimmte Ähnlichkeit vom entsprechend gefundenen Minimum abhängt, welches wie bereits erwähnt nicht das globale Minimum sein muss.

Ohne eine Ähnlichkeit der Punktmenge zum Modell wird das Modell häufig in Positionen transformiert, die relativ wahllos durch ein lokales Minimum bestimmt wurden. Daher ist es durch eine entsprechende Detektion (siehe Kapitel 3.4) nötig die Datenmenge zuvor so einzugrenzen, dass möglichst viele Punkte des zu registrierenden Objekts und wenig Punkte vom Hintergrund in der Datenmenge vorhanden sind.

D.1. Iterative Closest Point - Algorithmus

Es gibt unterschiedlichste Variationen des ICP-Algorithmus Rusinkiewicz u. Levoy [2001]. Die meisten lassen sich in sechs Schritte aufteilen:

1. Auswahl der Punkte in den Punktwolken (Modell und Daten)
2. Finden der Korrespondenzen zwischen dem Modell und den Daten
3. Gewichtung der Korrespondenzen
4. Ausschluss von Punktkorrespondenzen
5. Bestimmen des Fehlers
6. Minimieren des Fehlers

In dem in dieser Arbeit verwendeten Algorithmus wird keine Gewichtung durchgeführt, wodurch Schritt 3 komplett wegfällt. Zudem lassen sich Schritt 5 und 6 als ein Schritt betrachten, da die Minimierung des Fehlers diesen als Ergebnis liefert. Die verwendete Version führt zudem keine spezielle Auswahl der Punkte aus, wodurch auch Schritt 1 wegfällt.

Die verwendete Version lässt sich somit in 3 Unterschritte aufteilen:

1. Finden der Korrespondenzen zwischen dem Modell und den Daten
2. Ausschluss von Punktkorrespondenzen
3. Bestimmen und Minimieren des Fehlers

Aus Gründen des Umfangs dieser Arbeit werden die einzelnen Schritte nur kurz erläutert und auf entsprechende Literatur verwiesen. Eine Einleitung kann in der Arbeit von Padia u. Pears [2011] gefunden werden, an der sich im Folgenden orientiert wurde.

Im **ersten Schritt** wurde ein k-d Baum (Abkürzung für k-dimensionaler Binärbaum) verwendet, wie es in der Arbeit von Besl u. McKay [1992] vorgeschlagen wird. Genauer wird eine Delaunay-Triangulation [O'Rourke 1998; Aurenhammer 1991] generiert. Diese ist in der Bibliothek QHull verfügbar und wird in [Barber u. a. 1996] vorgestellt. Die QHull-Bibliothek wird ebenfalls in MATLAB[®] eingesetzt. Durch inkrementelles Hinzufügen einzelner Punkte zur Triangulation kann ein entsprechender k-d Baum erzeugt werden [Teillaud 1993]. Durch das Generieren eines entsprechenden Baumes für eine der beiden 3D Punktwolken, können die Punktkorrespondenzen zwischen den beiden 3D Punktwolken effizient bestimmt werden. Dabei werden solche Punkte verknüpft, welche die minimale Entfernung zueinander haben.

Anschließend werden im **zweiten Schritt** solche Punkte aus beiden Punktwolken ausgeschlossen, deren Entfernung zwischen den korrespondierenden Punkten zu groß

ist. Hierzu wird ein Schwellwert festgelegt, der eine entsprechende Selektion erlaubt.

Im **dritten Schritt** geht es darum die Punkte $\mathbf{x}^{\mathbf{m}}$ des Modells M mit $\mathcal{P} : \mathbf{x}^{\mathbf{m}} \in \mathbb{R}^3 \mapsto \hat{\mathbf{x}}^{\mathbf{m}} := R \mathbf{x}^{\mathbf{m}} + \mathbf{t} \in \mathbb{R}^3$ so zu transformieren, dass sie bestmöglich mit den Punkten $\mathbf{x}^{\mathbf{p}}$ der 3D Punktwolke P , auf die das Modell zu registrieren ist, übereinstimmen. Die Transformation besteht dabei aus einer Rotation $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ und einer Translation $\mathbf{t} \in \mathbb{R}^3$, welche sechs Freiheitsgrade besitzt (jeweils 3 für die Rotation und die Translation). Ziel ist es, mit der passenden Transformation die Differenzen der Punktkorrespondenzen

$$\begin{aligned} d_k &= \frac{1}{N_k} \sum_{i=1}^{N_k} \|\mathbf{x}^{\mathbf{p}}_{i,k} - \hat{\mathbf{x}}^{\mathbf{m}}_{i,k}\|^2 = \frac{1}{N_k} \sum_{i=1}^{N_k} \|\mathbf{x}^{\mathbf{p}}_{i,k} - \mathcal{P}(\mathbf{x}^{\mathbf{m}}_{i,k})\|^2 \\ &= \frac{1}{N_k} \sum_{i=1}^{N_k} \|\mathbf{x}^{\mathbf{p}}_{i,k} - \mathbf{R} \mathbf{x}^{\mathbf{m}}_{i,k} - \mathbf{t}\|^2, \end{aligned} \quad (\text{D.1})$$

welche den entsprechenden Fehlerterm darstellt, zu minimieren. N_k ist die Anzahl der in den Schritten 1 & 2 bestimmten Punktkorrespondenzen $\mathbf{x}^{\mathbf{p}}_{i,k} \in P$ bzw. $\mathbf{x}^{\mathbf{m}}_{i,k} \in M$. Wie in [Besl u. McKay 1992] für das Registrieren von 3D Punkten empfohlen, wird hier die Methode der Quaternionen von Horn [1987] verwendet. Diese schließt, im Gegensatz zur Methode über die Singulärwertzerlegung, Reflexionen als mögliche Lösung aus.

Die Methode der Quaternionen minimiert die Fehlerfunktion

$$f(\mathbf{q}) = \frac{1}{N_k} \sum_{i=1}^{N_k} N_k \|\mathbf{x}^{\mathbf{p}}_{i,k} - \mathbf{R}(\mathbf{q}) \mathbf{x}^{\mathbf{m}}_{i,k} - \mathbf{t}_q\|^2. \quad (\text{D.2})$$

Deren Ähnlichkeit zu Gleichung D.1 ist unverkennbar. $\mathbf{R}(\mathbf{q}) \in \mathbb{R}^{3 \times 3}$ und $\mathbf{t}_q \in \mathbb{R}^3$ sind eine Rotationsmatrix und ein Translationsvektor. Diese können nach Horn [1987] aus \mathbf{q} bestimmt werden. $\mathbf{q} = [q_0 \ q_1 \ q_2 \ q_3]^T \in \mathbb{R}^4$ ist die Vektordarstellung der zu bestimmenden Einheitsquaternion \hat{q} . Die Einheitsquaternion kann als eine Zahl mit einem Realteil und drei Imaginärteilen (i, j und k) aufgeschrieben werden

$$\hat{q} = q_0 + i \ q_1 + j \ q_2 + k \ q_3. \quad (\text{D.3})$$

Mit

$$\hat{q} = \cos\left(\frac{\theta}{2}\right) + \sin\left(\frac{\theta}{2}\right) (i \ w_x + j \ w_y + k \ w_z) \quad (\text{D.4})$$

wird durch die Einheitsquaternion \hat{q} eine Rotation um den Einheitsvektor $\mathbf{w} = [w_x \ w_y \ w_z]^T \in \mathbb{R}^3$ mit dem Winkel θ repräsentiert.

Die Transformation \mathbf{t}_q aus Gleichung D.2 ist durch

$$\mathbf{t}_q = \overline{\mathbf{x}^{\mathbf{p}}} - \mathbf{R}(\mathbf{q}) \overline{\mathbf{x}^{\mathbf{m}}} \quad (\text{D.5})$$

zu berechnen. $\overline{\mathbf{x}^{\mathbf{p}}}$ bzw. $\overline{\mathbf{x}^{\mathbf{m}}}$ sind die Zentren der Punktwolken P bzw. M und werden

D. Registrieren von 3D Daten durch den Iterative Closest Point - Algorithmus

durch

$$\overline{\mathbf{x}^{\mathbf{p}}} = \frac{1}{N_k} \sum_{i=1}^{N_k} \mathbf{x}^{\mathbf{p}}_{i,k} \quad bzw. \quad \overline{\mathbf{x}^{\mathbf{m}}} = \frac{1}{N_k} \sum_{i=1}^{N_k} \mathbf{x}^{\mathbf{m}}_{i,k} \quad (\text{D.6})$$

bestimmt.

Der Vektor \mathbf{q} der Einheitsquaternion ist nach Horn [1987] der Eigenvektor, welcher zum größten Eigenwert der symmetrischen 4x4 Matrix

$$\mathbf{N} = \begin{bmatrix} (S_{xx} + S_{yy} + S_{zz}) & S_{yz} - S_{zy} & S_{zx} - S_{xz} & S_{xy} - S_{yx} \\ S_{yz} - S_{zy} & (S_{xx} - S_{yy} - S_{zz}) & S_{xy} + S_{yx} & S_{zx} + S_{xz} \\ S_{zx} - S_{xz} & S_{xy} + S_{yx} & (-S_{xx} + S_{yy} - S_{zz}) & S_{yz} + S_{zy} \\ S_{xy} - S_{yx} & S_{zx} + S_{xz} & S_{yz} + S_{zy} & (-S_{xx} - S_{yy} + S_{zz}) \end{bmatrix} \quad (\text{D.7})$$

gehört. Die Elemente der Matrix \mathbf{N} sind zusammengestellt aus der Kreuzkovarianzmatrix

$$\begin{aligned} \mathbf{M} &= \begin{pmatrix} S_{xx} & S_{xy} & S_{xz} \\ S_{yx} & S_{yy} & S_{yz} \\ S_{zx} & S_{zy} & S_{zz} \end{pmatrix} \\ &= \sum_{i=1}^{N_k} (\mathbf{x}^{\mathbf{m}}_{i,k} - \overline{\mathbf{x}^{\mathbf{m}}_{i,k}}) (\mathbf{x}^{\mathbf{p}}_{i,k} - \overline{\mathbf{x}^{\mathbf{p}}_{i,k}})^T = \sum_{i=1}^{N_k} \mathbf{x}^{\mathbf{m}}_{i,k} \mathbf{x}^{\mathbf{p}^T}_{i,k} - N_k \overline{\mathbf{x}^{\mathbf{m}}_{i,k}} \overline{\mathbf{x}^{\mathbf{p}}_{i,k}}^T. \end{aligned} \quad (\text{D.8})$$

In der Arbeit [Horn 1987] ist beschrieben, dass sich die Rotationsmatrix $\mathbf{R}(\mathbf{q})$ aus der bestimmten Einheitsquaternion $\mathbf{q} = [q_0 \ q_1 \ q_2 \ q_3]^T$ durch

$$\mathbf{R}(\mathbf{q}) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 - q_0 q_2) \\ 2(q_1 q_2 - q_0 q_3) & q_0^2 + q_2^2 - q_1^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 - q_0 q_1) & q_0^2 + q_3^2 - q_1^2 - q_2^2 \end{bmatrix} \quad (\text{D.9})$$

ergibt.

Die Methode der Quaternionen liefert für die zuvor festgelegten Punktkorrespondenzen eine geschlossene Lösung.

Der ICP-Algorithmus ruft iterativ die Schritte 1-3 auf, wobei sich in jedem Iterationsschritt k jeweils andere Punktkorrespondenzen ergeben, da sich das Modell M immer weiter an die Punktwolke P annähert. Beendet wird dieses Vorgehen, sobald die Änderung des Fehlers d_k (siehe Gleichung D.1) unterhalb einer vordefinierten Grenze δ liegt

$$d_k - d_{k+1} < \delta \quad . \quad (\text{D.10})$$

d_{k+1} ist der Fehler des Folgeschritts $k+1$ vom Iterationsschritt k , dessen Fehler d_k ist. Die Punkte des Modells $\mathbf{x}^{\mathbf{m}} \in M$ verändern sich somit in jedem Iterationsschritt $\mathbf{x}^{\mathbf{m}}_{k+1} = \mathbf{R}_k \mathbf{x}^{\mathbf{m}}_k + \mathbf{t}_k$. \mathbf{R}_k und \mathbf{t}_k sind die Rotationsmatrix bzw. der Translationsvektor aus dem k -ten Iterationsschritt. Während des gesamten Verfahrens werden

D.1. Iterative Closest Point - Algorithmus

Rotation und Translation mit $\mathbf{R}_{k+1} = \mathbf{R}_k \mathbf{R}_{k-1}$ und $\mathbf{t}_{k+1} = \mathbf{R}_k \mathbf{t}_{k-1} + \mathbf{t}_k$ durch die aktuellen Rotations- \mathbf{R}_k und Translationsschritte \mathbf{t}_k laufend aktualisiert.

E. Ergebnisse der Detektion

E.1. Parametervergleich

E.1.1. Parametervergleich I

Die Tabellen E.1-E.7 enthalten die nach der *Parameteroptimierung I* aus Kapitel 3.7.5 optimierten Parameter zur Segmentierung nach Kapitel 3.4.3. Jede Tabelle steht für einen Parametersatz (nach Tabelle 3.6 aus Kapitel 3.7.5), die einzelnen Zeilen in den Tabellen entsprechen den Lerndatensätzen, mit denen die Parameter optimiert wurden. Die Gesamtkosten der Lern- c_I bzw. Testdatensätze \hat{c}_I in den beiden letzten Spalten, der hier aufgeführten Tabellen E.1-E.7, sind entsprechend der Tabelle 4.7 aus Kapitel 4.4.1 farblich markiert.

Tabelle E.1.: Parameter (1) der Segmentierung aus Kapitel 3.4.3

#	Parametersatz $\{T_\psi\}$				
	α	t_b	T_ψ	c_I	\hat{c}_I
1	50,7°	11,3	6,6°	105,3	108,8
2	50,4°	32,0	3,8°	102,3	102,2
3	47,5°	11,3	6,3°	107,9	110,7
4	50,4°	6,4	8,4°	105,0	109,6
5	49,3°	32,0	4,6°	98,7	107,3
6	46,5°	10,3	8,6°	100,7	100,2
7	48,2°	10,3	7,2°	106,4	105,3
8	52,4°	10,5	4,8°	107,2	112,1
9	51,1°	10,5	4,2°	104,7	106,0
10	47,0°	8,8	8,4°	99,6	103,3
11	47,4°	10,5	6,5°	98,0	104,8
12	47,4°	32,0	3,8°	106,5	105,2
13	46,7°	11,3	7,3°	106,9	101,7
14	46,6°	10,5	7,0°	103,5	105,5
15	48,5°	32,0	4,3°	104,3	110,8

Tabelle E.2.: Parameter (2) der Segmentierung aus Kapitel 3.4.3

Parametersatz $\{T_\theta\}$					
#	α	t_b	T_θ	c_I	\hat{c}_I
1	58,8°	4,4	7,2°	76,0	76,2
2	58,6°	6,3	6,1°	72,6	75,8
3	59,6°	4,3	7,5°	74,8	76,3
4	58,3°	4,4	6,6°	74,5	75,6
5	58,1°	3,6	8,7°	73,2	74,9
6	57,2°	4,2	7,5°	73,8	75,2
7	57,4°	5,6	6,6°	74,3	77,7
8	58,7°	5,0	6,7°	76,4	78,9
9	58,9°	3,4	8,9°	75,3	75,8
10	55,5°	4,9	6,9°	74,7	77,3
11	56,4°	5,4	6,4°	74,1	75,6
12	58,8°	4,0	8,4°	71,8	78,3
13	59,7°	3,1	10,1°	76,3	74,3
14	57,8°	5,4	7,1°	75,4	74,2
15	57,4°	4,4	7,5°	75,2	75,8

Tabelle E.3.: Parameter (3) der Segmentierung aus Kapitel 3.4.3

Parametersatz $\{T_\delta\}$					
#	α	t_b	T_δ	c_I	\hat{c}_I
1	58,7°	24,0	2,8 cm	79,1	83,2
2	58,3°	24,0	3,0 cm	80,7	80,6
3	60,1°	13,5	2,8 cm	79,6	79,1
4	58,2°	24,0	2,7 cm	80,6	83,0
5	58,0°	24,0	2,9 cm	79,3	81,4
6	56,6°	24,0	2,9 cm	77,8	80,3
7	59,6°	24,0	2,6 cm	79,7	82,8
8	59,9°	24,0	2,9 cm	79,4	87,0
9	58,9°	24,0	2,9 cm	83,8	80,1
10	59,2°	32,0	2,9 cm	78,3	80,2
11	55,9°	24,0	2,8 cm	75,9	83,4
12	57,2°	24,0	3,1 cm	75,5	81,9
13	58,9°	13,5	2,8 cm	81,2	74,9
14	60,3°	24,0	2,7 cm	80,1	78,9
15	57,3°	24,0	2,7 cm	80,9	83,1

Tabelle E.4.: Parameter (4) der Segmentierung aus Kapitel 3.4.3

Parametersatz $\{T_\psi, T_\theta\}$						
#	α	t_b	T_ψ	T_θ	c_I	\hat{c}_I
1	58,2°	10,5	7,1°	6,1°	75,3	74,9
2	56,7°	10,5	5,3°	7,0°	74,1	76,2
3	59,5°	7,5	10,8°	5,4°	74,0	73,6
4	57,3°	10,5	13,2°	4,5°	74,1	73,4
5	57,1°	7,4	14,7°	5,0°	73,5	75,3
6	57,6°	7,4	13,5°	4,5°	73,5	74,1
7	58,1°	8,0	7,0°	6,3°	73,4	77,3
8	59,5°	8,8	8,9°	5,6°	74,0	77,0
9	58,6°	9,3	9,7°	4,9°	75,1	74,4
10	56,3°	7,6	9,6°	6,5°	73,8	77,7
11	55,0°	9,3	12,0°	5,2°	73,3	76,5
12	58,3°	10,5	9,0°	5,3°	72,0	76,6
13	57,6°	10,4	7,1°	6,1°	75,7	74,5
14	57,5°	8,3	9,1°	6,0°	73,4	73,2
15	57,3°	10,5	6,7°	6,1°	73,8	75,7

Tabelle E.5.: Parameter (5) der Segmentierung aus Kapitel 3.4.3

Parametersatz $\{T_\psi, T_\delta\}$						
#	α	t_b	T_ψ	T_δ	c_I	\hat{c}_I
1	60,1°	24,0	3,8°	3,6 cm	80,8	83,0
2	58,3°	24,0	43,3°	3,0 cm	80,7	80,6
3	60,1°	13,5	44,7°	2,8 cm	79,6	79,1
4	58,9°	20,0	8,3°	2,8 cm	80,7	82,8
5	58,0°	24,0	40,0°	2,9 cm	79,3	81,4
6	58,7°	22,0	4,0°	3,1 cm	77,0	80,4
7	59,6°	24,0	44,6°	2,6 cm	79,8	82,8
8	59,9°	24,0	43,6°	2,9 cm	79,4	87,0
9	61,0°	24,0	3,3°	3,6 cm	83,7	81,3
10	59,2°	32,0	27,4°	2,9 cm	78,3	79,9
11	55,9°	24,0	35,6°	2,8 cm	75,8	83,2
12	58,3°	24,0	14,9°	3,1 cm	75,3	81,4
13	59,9°	13,5	15,7°	2,8 cm	81,4	75,9
14	60,0°	21,0	3,7°	3,4 cm	81,2	79,4
15	59,7°	25,0	3,3°	3,5 cm	81,8	81,6

Tabelle E.6.: Parameter (6) der Segmentierung aus Kapitel 3.4.3

Parametersatz $\{T_\theta, T_\delta\}$						
#	α	t_b	T_θ	T_δ	c_I	\hat{c}_I
1	58,7°	24,0	40,7°	2,8 cm	79,1	83,2
2	58,3°	24,0	35,0°	3,0 cm	80,9	80,8
3	60,1°	13,4	32,2°	2,8 cm	79,6	79,1
4	57,3°	20,0	3,6°	5,3 cm	76,6	77,3
5	58,0°	24,0	44,9°	2,9 cm	79,3	81,4
6	56,1°	24,0	2,9°	4,9 cm	74,9	75,3
7	56,9°	26,0	3,5°	4,3 cm	74,8	77,6
8	59,9°	24,0	43,7°	2,9 cm	79,4	87,0
9	58,0°	24,0	40,4°	2,7 cm	84,0	78,3
10	58,4°	32,0	4,4°	3,5 cm	75,6	77,9
11	55,9°	24,0	37,0°	2,8 cm	75,9	83,4
12	56,9°	24,0	3,2°	5,7 cm	73,6	78,1
13	56,6°	26,0	3,4°	5,3 cm	77,4	75,4
14	60,3°	24,0	44,2°	2,7 cm	80,1	78,8
15	58,3°	28,0	3,0°	4,5 cm	75,5	77,1

Tabelle E.7.: Parameter (7) der Segmentierung aus Kapitel 3.4.3

Parametersatz $\{T_\psi, T_\theta, T_\delta\}$							
#	α	t_b	T_ψ	T_θ	T_δ	c_I	\hat{c}_I
1	58,7°	24,0	13,7°	4,5°	3,7 cm	75,9	77,7
2	56,9°	24,0	4,9°	3,4°	10,9 cm	74,8	76,9
3	60,1°	13,3	18,1°	15,9°	2,7 cm	79,6	79,5
4	57,3°	20,0	4,6°	3,4°	8,2 cm	76,2	76,0
5	57,0°	14,0	8,8°	3,0°	8,2 cm	75,9	76,9
6	59,3°	28,0	3,4°	7,2°	3,6 cm	77,9	80,8
7	56,9°	32,0	43,6°	3,5°	4,3 cm	74,8	77,8
8	59,9°	24,0	43,2°	44,6°	2,9 cm	79,4	87,0
9	58,3°	14,0	10,6°	3,4°	4,1 cm	78,4	76,0
10	56,1°	32,0	6,2°	43,5°	3,6 cm	78,3	81,2
11	55,5°	24,0	28,6°	4,3°	3,6 cm	74,0	78,9
12	58,1°	24,0	8,3°	5,6°	3,6 cm	74,2	77,9
13	58,7°	13,0	11,3°	5,2°	3,6 cm	78,4	76,4
14	59,3°	24,0	6,6°	3,2°	7,6 cm	74,6	75,0
15	58,9°	27,0	43,3°	3,4°	3,6 cm	76,7	77,7

E.1.2. Parametervergleich II

Die Tabellen E.8-E.13 enthalten die optimierten Parameter (*Parameteroptimierung II* aus Kapitel 3.7.5) zur Detektion (Kapitel 3.4.4) nach dem Clustern aus Kapitel 3.4.2. Zum Clustern kamen die bereits für jeden Datensatz optimierten Parameter (*Parameteroptimierung I* in Kapitel 3.7.5) aus Tabelle 4.6 zum Einsatz. Jede Tabelle steht für einen Parametersatz (nach Tabelle 3.7 aus Kapitel 3.7.5). Die einzelnen Zeilen in den Tabellen entsprechen den Lerndatensätzen, mit denen die Parameter optimiert wurden. Die Gesamtkosten der Lern- c_{II} bzw. Testdatensätze \hat{c}_{II} in den beiden letzten Spalten, der hier aufgeführten Tabellen E.8-E.13, sind entsprechend der Tabelle 4.8 aus Kapitel 4.4.1 farblich markiert.

Tabelle E.8.: Parameter (1) der Detektion nach dem Clustern aus Kapitel 3.4.2

#	b_ψ	Parametersatz $\{b_\psi\}$	
		$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	8,1°	6,1	9,2
2	0,6°	16,7	20,1
3	10,0°	8,5	7,9
4	0,6°	21,3	27,4
5	34,8°	13,2	18,3
6	0,6°	20,9	17,9
7	42,9°	8,0	8,0
8	0,6°	20,5	21,7
9	0,8°	8,4	5,5
10	6,4°	27,1	24,7
11	8,2°	10,8	10,0
12	0,6°	9,8	12,6
13	8,3°	10,3	13,0
14	10,0°	13,5	14,0
15	0,6°	13,9	15,7

Tabelle E.9.: Parameter (2) der Detektion nach dem Clustern aus Kapitel 3.4.2

Parametersatz $\{b_\theta\}$			
#	b_θ	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	21,6°	16,7	19,4
2	6,8°	28,2	29,0
3	6,3°	19,7	18,6
4	20,9°	24,3	26,5
5	11,0°	24,4	30,9
6	40,6°	33,9	36,3
7	7,5°	17,1	20,5
8	40,3°	30,0	31,2
9	6,2°	17,2	16,0
10	22,9°	36,2	35,4
11	28,2°	19,4	19,8
12	28,5°	17,9	19,7
13	0,6°	23,5	25,9
14	29,9°	23,6	23,6
15	32,9°	21,3	28,7

Tabelle E.10.: Parameter (3) der Detektion nach dem Clustern aus Kapitel 3.4.2

Parametersatz $\{b_\delta\}$			
#	b_δ	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	6,3 <i>cm</i>	12,8	13,8
2	8,3 <i>cm</i>	22,8	26,8
3	11,2 <i>cm</i>	13,3	12,4
4	8,6 <i>cm</i>	22,2	28,8
5	7,3 <i>cm</i>	22,2	29,9
6	7,3 <i>cm</i>	31,0	33,1
7	10,0 <i>cm</i>	13,5	17,0
8	6,3 <i>cm</i>	35,3	45,6
9	9,4 <i>cm</i>	12,5	9,7
10	8,1 <i>cm</i>	32,6	35,4
11	7,9 <i>cm</i>	15,6	18,5
12	6,5 <i>cm</i>	16,7	20,0
13	7,5 <i>cm</i>	13,2	19,0
14	5,3 <i>cm</i>	20,4	25,2
15	8,1 <i>cm</i>	13,8	19,0

Tabelle E.11.: Parameter (4) der Detektion nach dem Clustern aus Kapitel 3.4.2

#	Parametersatz $\{b_\psi, b_\theta\}$			
	b_ψ	b_θ	$c_{II}[\%]$	$c_{\hat{II}}[\%]$
1	6,0°	17,8°	5,6	9,8
2	0,6°	44,2°	16,7	20,4
3	10,0°	44,8°	8,7	7,9
4	22,5°	25,2°	24,0	25,9
5	25,9°	40,4°	13,7	19,1
6	0,6°	44,9°	21,3	18,6
7	14,5°	12,7°	7,7	9,3
8	0,6°	44,9°	20,6	21,8
9	6,4°	5,7°	8,6	6,0
10	6,4°	44,9°	27,3	24,8
11	10,0°	38,5°	11,4	10,0
12	6,4°	23,0°	9,8	13,4
13	10,0°	42,7°	10,4	13,2
14	2,8°	28,8°	13,3	14,4
15	0,6°	44,5°	14,2	16,0

Tabelle E.12.: Parameter (5) der Detektion nach dem Clustern aus Kapitel 3.4.2

#	Parametersatz $\{b_\psi, b_\delta\}$			
	b_ψ	b_δ	$c_{II}[\%]$	$c_{\hat{II}}[\%]$
1	0,6°	14,9 cm	5,8	9,5
2	0,6°	8,4 cm	12,8	17,2
3	0,6°	14,5 cm	7,7	7,0
4	0,7°	14,9 cm	16,0	23,0
5	0,6°	14,9 cm	10,2	16,6
6	0,6°	14,9 cm	15,9	17,1
7	16,9°	14,9 cm	7,8	8,8
8	0,6°	14,9 cm	17,0	21,4
9	0,6°	13,7 cm	8,1	5,4
10	0,6°	14,9 cm	19,6	21,8
11	8,2°	14,5 cm	9,5	10,2
12	0,6°	14,9 cm	8,3	12,6
13	0,6°	9,0 cm	8,0	11,5
14	0,6°	13,9 cm	10,1	13,0
15	0,6°	14,9 cm	9,6	13,0

Tabelle E.13.: Parameter (6) der Detektion nach dem Clustern aus Kapitel 3.4.2

#	Parametersatz $\{b_\theta, b_\delta\}$			
	b_θ	b_δ	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	10,3°	10,0 cm	9,6	12,2
2	33,9°	8,3 cm	21,1	25,4
3	8,3°	9,4 cm	12,3	10,8
4	20,1°	7,7 cm	19,8	27,0
5	32,8°	7,5 cm	21,2	28,5
6	32,2°	7,3 cm	28,8	33,0
7	10,9°	9,2 cm	8,7	11,1
8	30,6°	6,3 cm	30,4	42,3
9	6,6°	11,4 cm	11,3	7,5
10	20,1°	8,2 cm	29,3	33,8
11	9,0°	8,6 cm	11,5	14,8
12	9,8°	9,2 cm	15,5	19,2
13	32,6°	7,3 cm	12,4	18,2
14	22,3°	8,6 cm	19,4	20,8
15	26,2°	7,5 cm	11,8	17,8

Die Tabellen E.14-E.19 enthalten die nach der *Parameteroptimierung II* aus Kapitel 3.7.5 optimierten Parameter der Detektion nach der Segmentierung aus Kapitel 3.4.3 mit den Parametern aus Tabelle E.4 (Parametersatz $\{T_\psi, T_\theta\}$). Jede Tabelle steht für einen Parametersatz, die einzelnen Zeilen in den Tabellen entsprechen den Lern- c_{II} bzw. Testdatensätze \hat{c}_{II} in den beiden letzten Spalten der hier aufgeführten Tabellen E.8-E.13 sind entsprechend der Tabelle 4.8 aus Kapitel 4.4.1 farblich markiert.

Tabelle E.14.: Parameter (1) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	Parametersatz $\{b_\psi\}$		
	b_ψ	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	33,3°	5,9	8,1
2	30,6°	6,5	7,5
3	28,8°	9,0	8,6
4	10,9°	7,5	7,0
5	29,7°	9,2	10,4
6	11,8°	9,9	10,3
7	28,8°	9,5	8,4
8	30,1°	7,4	9,9
9	30,1°	7,3	7,2
10	29,7°	7,4	11,4
11	27,9°	8,1	8,7
12	29,7°	7,4	7,1
13	29,7°	7,1	7,8
14	30,1°	8,8	8,9
15	30,1°	7,9	6,9

Tabelle E.15.: Parameter (2) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	Parametersatz $\{b_\theta\}$		
	b_θ	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	14,9°	4,8	6,4
2	17,6°	4,3	6,2
3	6,0°	5,2	5,2
4	18,1°	4,3	4,5
5	0,6°	4,6	5,8
6	44,0°	4,9	12,9
7	44,9°	3,8	6,5
8	44,9°	5,5	5,8
9	0,6°	4,4	5,3
10	16,7°	5,2	6,3
11	16,7°	4,9	5,5
12	0,6°	4,4	5,4
13	17,2°	5,4	5,2
14	14,5°	5,9	4,3
15	17,6°	5,0	6,0

E. Ergebnisse der Detektion

Tabelle E.16.: Parameter (3) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	Parametersatz $\{b_\delta\}$		
	b_δ	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	9,8 cm	15,6	19,9
2	9,0 cm	15,9	21,6
3	9,8 cm	16,2	18,9
4	5,9 cm	17,1	17,6
5	3,6 cm	23,7	23,4
6	1,2 cm	22,3	21,9
7	6,7 cm	19,0	20,9
8	4,3 cm	15,4	15,9
9	2,0 cm	17,5	16,5
10	5,1 cm	19,8	21,4
11	2,0 cm	23,1	26,1
12	5,9 cm	16,0	17,0
13	5,9 cm	16,2	17,9
14	1,2 cm	16,7	17,2
15	1,2 cm	19,1	16,5

Tabelle E.17.: Parameter (4) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	Parametersatz $\{b_\psi, b_\theta\}$			
	b_ψ	b_θ	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	18,5°	14,9°	0,8	2,2
2	40,0°	13,6°	0,3	1,5
3	10,9°	14,5°	1,9	2,0
4	10,0°	14,5°	0,8	0,8
5	8,7°	13,1°	0,7	2,2
6	7,3°	13,6°	1,7	1,7
7	41,3°	12,7°	2,0	1,5
8	38,6°	14,9°	1,7	2,2
9	10,9°	14,5°	1,2	1,5
10	14,5°	13,6°	1,1	2,5
11	10,9°	14,0°	1,0	2,3
12	7,3°	13,6°	0,9	2,0
13	13,1°	13,1°	0,7	1,7
14	42,7°	13,6°	1,6	1,0
15	44,9°	14,9°	0,5	1,3

Tabelle E.18.: Parameter (5) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	Parametersatz $\{b_\psi, b_\delta\}$			
	b_ψ	b_δ	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	5,5°	13,7 cm	1,4	3,0
2	6,4°	7,5 cm	0,9	3,2
3	0,6°	12,9 cm	2,7	3,0
4	7,3°	11,4 cm	1,4	1,2
5	0,6°	11,4 cm	1,3	3,3
6	0,6°	11,4 cm	2,5	2,4
7	1,5°	10,6 cm	3,3	3,9
8	1,9°	11,4 cm	1,8	3,1
9	0,6°	11,4 cm	1,5	1,7
10	5,5°	9,0 cm	2,3	4,7
11	1,9°	11,4 cm	1,6	2,9
12	7,3°	12,2 cm	0,8	2,2
13	7,3°	11,4 cm	1,1	2,4
14	7,3°	11,4 cm	2,3	1,7
15	7,3°	12,2 cm	1,3	1,5

Tabelle E.19.: Parameter (6) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	Parametersatz $\{b_\theta, b_\delta\}$			
	b_θ	b_δ	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	14,9°	14,5 cm	0,9	2,5
2	14,9°	14,5 cm	0,7	1,8
3	16,3°	12,9 cm	1,9	1,9
4	14,9°	12,9 cm	1,1	1,0
5	6,4°	11,4 cm	1,1	2,9
6	12,2°	12,2 cm	2,0	1,9
7	14,0°	12,9 cm	2,2	1,8
8	12,7°	12,2 cm	1,5	2,2
9	11,8°	11,4 cm	1,3	1,5
10	12,7°	12,2 cm	2,0	2,5
11	13,1°	12,2 cm	1,1	2,3
12	16,7°	14,5 cm	0,8	2,0
13	13,1°	12,2 cm	1,0	2,5
14	14,5°	12,9 cm	1,3	1,1
15	14,0°	14,5 cm	1,0	1,1

E. Ergebnisse der Detektion

Die Tabellen E.20-E.25 sind die verbleibenden Kombinationen der *Parameteroptimierung I* und *Parameteroptimierung II* aus Kapitel 3.7.5. Sie entsprechen der Tabelle 4.9 aus Kapitel 4.4.1. Die entsprechend optimierten Parameter der Detektion, wie sie in den Tabellen E.14-E.19 für die Parameterkombination $\{T_\psi, T_\theta\}$ der Segmentierung aus Kapitel 3.4.3 aufgeführt sind, werden wegen der Menge der Daten (insgesamt 42 Tabellen) nicht weiter angegeben.

Tabelle E.20.: Ergebnisse (1) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	{ b_ψ }		{ b_θ }		{ b_δ }		{ b_ψ, b_θ }		{ b_ψ, b_δ }		{ b_θ, b_δ }	
	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	11,4	18,0	15,6	12,2	36,5	41,3	6,7	5,6	10,1	15,7	5,3	5,6
2	15,7	16,0	14,4	10,4	39,2	34,3	12,1	11,7	11,6	13,7	9,0	5,1
3	18,5	20,3	11,9	7,7	41,8	41,2	11,2	7,1	20,7	21,2	14,0	9,6
4	12,4	17,0	7,5	5,2	38,3	38,2	6,2	5,4	12,0	15,2	7,0	4,9
5	12,0	16,7	13,9	12,6	40,1	39,0	5,5	6,2	13,0	15,6	6,9	6,1
6	13,9	14,7	15,6	13,0	44,4	40,6	9,7	8,2	17,9	16,6	11,8	8,6
7	13,9	14,6	10,4	9,0	42,4	37,8	10,1	6,5	16,2	16,7	14,1	8,4
8	12,9	20,3	7,5	7,3	38,2	34,4	7,8	9,7	11,4	16,4	6,9	5,4
9	14,7	15,3	8,4	5,8	41,5	35,6	7,6	6,0	13,9	14,7	7,9	4,1
10	14,3	16,5	10,3	16,1	42,0	37,1	8,9	6,5	17,2	17,2	11,7	9,2
11	14,8	18,1	15,3	14,8	40,0	40,3	7,3	8,5	17,2	21,2	9,6	10,6
12	15,0	15,4	16,9	13,1	42,4	39,3	8,9	10,0	16,1	17,4	11,2	7,6
13	17,9	17,4	17,3	11,6	41,7	36,7	10,2	6,3	18,5	17,6	12,6	7,8
14	17,3	19,1	17,3	14,3	41,5	37,0	10,2	8,6	17,5	19,2	13,0	10,1
15	15,1	17,5	17,3	16,5	41,8	38,0	7,8	7,6	11,9	14,7	10,7	7,8

Tabelle E.21.: Ergebnisse (2) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	{ b_ψ }		{ b_θ }		{ b_δ }		{ b_ψ, b_θ }		{ b_ψ, b_δ }		{ b_θ, b_δ }	
	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	10,3	13,1	6,1	8,8	21,4	24,1	1,5	3,8	2,7	4,5	1,4	3,1
2	10,3	12,5	4,7	5,5	19,3	23,2	0,5	1,5	2,2	4,9	0,7	1,6
3	11,3	10,2	6,4	5,7	20,0	21,3	2,8	3,2	3,4	4,2	2,2	2,0
4	12,0	11,0	5,5	5,2	21,6	21,9	1,9	1,5	2,9	2,2	2,3	1,5
5	8,8	9,3	6,7	7,0	25,0	24,6	1,8	2,0	2,2	3,0	1,4	1,5
6	10,1	11,6	5,3	6,1	25,7	26,4	1,4	2,4	2,0	3,2	1,5	2,4
7	10,9	10,1	6,3	6,4	25,1	25,5	3,1	2,7	4,9	4,9	3,0	3,2
8	8,4	11,9	5,9	6,3	19,9	18,7	3,0	3,3	4,2	5,0	2,3	2,9
9	9,5	10,4	5,4	7,0	21,2	20,4	2,0	2,8	3,3	3,4	2,0	2,1
10	8,5	10,4	5,3	6,4	22,7	22,3	1,8	1,9	3,1	3,9	2,1	2,1
11	11,8	11,5	5,4	6,1	25,9	26,7	1,9	2,4	3,0	3,4	1,6	2,2
12	10,3	10,7	4,1	5,9	19,4	19,7	1,1	3,4	1,6	4,2	1,3	2,4
13	9,9	10,0	6,0	6,2	19,1	17,8	1,9	2,2	2,1	2,5	1,1	1,6
14	11,7	12,8	5,7	5,0	20,2	20,5	2,7	2,2	4,2	4,3	1,9	2,1
15	12,4	11,4	7,2	7,4	24,3	21,1	2,0	2,6	3,3	3,0	1,7	1,9

Tabelle E.22.: Ergebnisse (4) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	Parametersatz (3) der Segmentierung: $\{T_\delta\}$											
	$\{b_\psi\}$		$\{b_\theta\}$		$\{b_\delta\}$		$\{b_\psi, b_\theta\}$		$\{b_\psi, b_\delta\}$		$\{b_\theta, b_\delta\}$	
	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	7,8	11,7	4,8	7,4	16,5	20,9	1,5	3,8	3,7	6,4	1,7	3,6
2	10,8	11,4	4,4	3,9	20,0	23,0	2,7	2,4	5,2	5,7	2,7	2,3
3	9,0	9,1	6,6	4,0	20,3	21,5	2,7	2,1	4,9	5,1	2,3	1,5
4	9,8	11,3	5,4	6,2	18,1	21,4	3,0	3,5	5,1	6,7	3,1	3,0
5	9,0	9,5	3,3	4,4	18,3	19,7	1,5	2,5	3,1	4,9	1,9	2,1
6	9,1	10,0	4,1	4,3	22,0	21,0	2,4	3,0	4,6	5,8	2,7	2,8
7	9,1	8,7	5,6	4,8	17,3	18,8	2,7	2,2	3,6	4,4	3,2	2,0
8	9,5	13,1	6,0	7,0	18,3	18,8	2,9	3,6	3,6	6,8	2,7	3,6
9	10,8	10,5	4,8	3,4	22,0	17,2	2,7	2,1	5,4	3,9	3,0	1,6
10	8,3	9,9	5,4	4,9	18,0	17,6	2,2	2,6	2,9	4,4	2,4	1,7
11	8,7	10,0	3,5	5,1	19,8	24,4	0,8	3,3	3,0	6,9	1,3	4,2
12	10,1	11,8	3,8	5,1	18,6	21,1	1,7	3,3	3,2	5,8	1,7	2,7
13	11,9	10,0	6,7	2,9	22,1	19,8	2,4	1,3	5,8	3,5	2,8	1,4
14	10,5	10,9	6,8	5,3	18,6	15,8	3,5	2,5	5,1	4,1	2,8	2,4
15	10,8	11,3	4,6	6,0	21,3	19,7	2,6	2,9	5,1	4,8	2,8	2,6

Tabelle E.23.: Ergebnisse (5) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	Parametersatz (5) der Segmentierung: $\{T_\psi, T_\delta\}$											
	$\{b_\psi\}$		$\{b_\theta\}$		$\{b_\delta\}$		$\{b_\psi, b_\theta\}$		$\{b_\psi, b_\delta\}$		$\{b_\theta, b_\delta\}$	
	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	8,4	12,3	4,8	7,5	15,9	18,5	1,5	3,4	4,2	6,3	2,7	4,8
2	10,9	11,3	4,4	3,8	20,1	22,9	2,8	2,4	5,2	5,7	2,8	2,2
3	9,0	9,1	6,6	4,0	20,3	21,5	2,7	2,1	4,9	5,1	2,3	1,5
4	10,2	11,0	5,2	6,1	15,0	17,5	2,4	2,9	4,3	5,5	3,0	4,2
5	9,0	9,5	3,3	4,5	18,3	19,7	1,6	2,6	3,1	4,9	1,9	2,1
6	9,7	10,4	3,9	5,9	14,4	14,5	1,1	2,1	2,5	3,8	1,3	2,9
7	9,2	8,8	5,6	4,8	17,3	18,8	2,7	2,2	3,6	4,4	3,3	2,0
8	9,4	13,0	6,0	7,0	18,2	18,8	2,9	3,6	3,6	6,8	3,5	4,9
9	10,6	8,9	5,5	4,9	18,0	13,8	3,1	1,8	5,8	4,1	4,3	3,0
10	8,3	9,4	5,5	4,9	18,4	17,4	2,2	2,4	3,2	4,2	2,4	1,7
11	8,6	9,8	3,4	5,0	19,4	24,3	0,8	3,3	3,0	7,0	1,3	4,2
12	9,8	11,2	3,8	4,8	14,6	15,0	1,4	2,0	2,5	4,2	1,2	1,4
13	11,7	10,6	7,1	3,2	18,8	16,6	2,1	1,1	5,2	3,3	2,3	1,8
14	11,0	12,2	6,6	5,2	15,6	13,5	3,7	2,8	5,3	4,8	4,2	3,4
15	10,8	11,4	4,9	5,6	16,9	15,5	2,2	2,4	4,4	3,9	3,1	3,2

Tabelle E.24.: Ergebnisse (6) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	Parametersatz (6) der Segmentierung: $\{T_\theta, T_\delta\}$											
	$\{b_\psi\}$		$\{b_\theta\}$		$\{b_\delta\}$		$\{b_\psi, b_\theta\}$		$\{b_\psi, b_\delta\}$		$\{b_\theta, b_\delta\}$	
	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	7,9	11,7	4,9	7,5	16,5	20,9	1,5	3,8	3,7	6,4	1,7	3,6
2	10,9	11,3	4,5	4,1	20,2	23,1	2,8	2,4	5,4	5,7	2,8	2,3
3	8,9	9,1	6,7	4,0	20,4	21,5	2,7	2,1	5,0	5,1	2,4	1,7
4	9,3	9,3	5,9	4,7	22,1	23,1	2,4	2,8	3,8	3,5	2,3	2,0
5	9,0	9,6	3,3	4,5	18,3	19,8	1,6	2,6	3,1	4,9	1,9	2,1
6	9,2	9,3	4,0	6,3	21,7	21,7	1,1	1,7	1,7	2,6	1,4	2,3
7	8,8	7,4	4,4	6,9	22,6	23,4	2,4	1,7	3,4	4,0	2,7	1,7
8	9,6	13,0	5,9	7,0	18,1	18,8	2,9	3,6	3,6	6,8	3,5	4,8
9	10,7	9,0	5,4	4,2	22,4	16,6	3,8	1,6	6,6	3,6	4,0	1,7
10	7,6	10,3	4,9	7,7	18,1	20,3	0,8	1,9	1,6	3,8	1,3	2,9
11	8,7	9,8	3,5	5,1	19,9	24,4	0,7	3,3	3,0	6,9	1,3	4,2
12	9,6	9,3	5,6	6,3	22,5	23,3	1,3	2,3	2,1	3,5	1,3	1,8
13	9,7	10,5	5,4	5,0	21,8	22,3	1,4	2,0	2,7	3,5	1,5	2,2
14	10,4	11,0	6,7	5,2	18,6	15,6	3,5	2,4	5,1	4,1	2,8	2,4
15	10,5	9,9	5,7	6,6	18,7	17,8	1,4	2,0	2,1	2,6	1,5	1,9

Tabelle E.25.: Ergebnisse (7) der Detektion nach der Segmentierung aus Kapitel 3.4.3

#	Parametersatz (7) der Segmentierung: $\{T_\psi, T_\theta, T_\delta\}$											
	$\{b_\psi\}$		$\{b_\theta\}$		$\{b_\delta\}$		$\{b_\psi, b_\theta\}$		$\{b_\psi, b_\delta\}$		$\{b_\theta, b_\delta\}$	
	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
1	7,4	10,2	4,1	6,3	16,1	19,1	0,8	2,8	1,8	4,0	1,0	3,5
2	7,9	8,6	4,7	5,3	17,1	23,3	1,2	2,3	2,0	3,2	1,6	2,4
3	8,4	8,0	6,3	4,8	12,6	13,6	2,3	1,4	2,6	1,9	1,9	1,1
4	8,8	8,4	5,7	5,5	17,9	19,0	2,0	2,4	3,0	3,5	2,1	2,5
5	9,5	9,6	5,9	6,4	25,6	23,7	1,4	2,1	1,9	3,1	1,7	2,8
6	9,2	10,5	3,7	5,6	15,7	16,2	1,9	2,5	2,8	4,5	2,2	3,6
7	8,8	7,3	4,4	6,9	22,3	23,5	2,4	1,7	3,4	3,9	2,7	1,7
8	9,4	13,0	6,0	7,0	18,2	18,8	2,9	3,7	3,6	6,8	3,5	4,9
9	9,5	7,7	6,3	5,7	21,3	17,0	2,6	1,1	4,3	2,5	3,2	2,0
10	7,2	10,7	4,7	5,2	14,0	15,2	2,2	2,8	3,5	5,6	2,9	4,1
11	9,1	9,7	3,8	4,9	22,1	25,2	0,4	2,1	2,0	4,8	1,0	4,0
12	8,9	9,3	4,0	4,6	16,1	17,6	1,2	1,7	1,9	3,0	1,4	1,5
13	10,8	11,1	5,0	3,5	12,9	13,5	1,8	1,1	3,0	3,3	2,1	1,9
14	9,4	10,0	5,4	5,1	13,9	14,7	1,8	2,1	2,4	2,9	1,8	2,1
15	10,0	9,8	5,3	6,6	18,1	17,3	1,3	2,1	2,2	2,7	1,6	1,9

E.2. Parameterwahl

E.2.1. Parameterwahl I

Die Tabellen E.26-E.28 listen die Ergebnisse der Kostenfunktionen c_I bzw. \hat{c}_I (Gleichung 3.111 aus Kapitel 3.7.5) auf, welche mit dem Minimum, dem Median, dem arithmetischen Mittel \varnothing und dem Maximum der 15 einzeln optimierten Parametersätze (siehe Kapitel 3.7.5) für die Lern- bzw. Testdatensätze mit der Segmentierung aus Kapitel 3.4.3 bestimmt wurden. Die entsprechenden Parameter sind in Tabelle 4.14 im Kapitel 4.4.3 aufgeführt. Für jeden Datensatz wurden jeweils die zwei besten Ergebnisse farblich markiert (bestes Ergebnis c_I in Blau, zweitbestes Ergebnis c_I in Cyan, bestes Ergebnis \hat{c}_I in Rot und das zweitbeste Ergebnis \hat{c}_I in Magenta). Die Anzahl N_{c_I} bzw. $N_{\hat{c}_I}$ in der Tabelle 4.14 aus Kapitel 4.4.3 entsprechen der Anzahl der farblich hervorgehobenen Datensätze in den Tabellen E.26-E.28 (beste und zweitbeste Ergebnisse).

Tabelle E.26 zeigt die Ergebnisse der ersten fünf Datensätze.

Tabelle E.26.: Ergebnisübersicht (1) der Segmentierung

		#1		#2		#3		#4		#5	
		c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I
$\{\alpha, t_b, T_\psi\}$	Min	123,5	123,2	122,8	120,4	126,7	127,0	122,1	127,0	118,9	121,8
	Med	107,6	107,2	105,0	103,1	108,4	110,2	105,8	110,2	99,9	106,0
	\varnothing	107,5	107,9	104,6	103,5	109,7	110,8	106,8	111,9	99,8	107,3
	Max	108,3	108,8	106,2	102,0	112,1	111,7	108,2	110,8	102,3	106,1
$\{\alpha, t_b, T_\theta\}$	Min	96,6	96,1	98,9	95,9	98,1	99,8	99,5	96,1	96,0	99,8
	Med	76,1	76,1	73,2	75,2	75,8	74,0	74,8	73,9	74,1	75,5
	\varnothing	77,0	75,8	72,8	75,1	75,3	75,5	75,8	73,5	74,3	74,9
	Max	88,6	85,6	81,7	87,5	85,5	84,0	86,1	83,7	83,8	81,2
$\{\alpha, t_b, T_\delta\}$	Min	81,6	84,2	84,3	83,1	81,7	81,9	83,6	83,3	81,3	82,1
	Med	79,1	83,2	81,5	80,6	80,6	79,3	81,5	83,1	80,1	81,7
	\varnothing	80,3	83,7	81,9	80,5	81,3	79,5	81,2	83,0	80,0	81,8
	Max	80,2	83,1	82,0	80,7	80,7	79,9	83,0	84,5	81,8	83,9
$\{\alpha, t_b, T_\psi, T_\theta\}$	Min	95,2	96,1	97,7	96,9	95,1	94,6	96,3	93,8	92,4	93,6
	Med	76,0	74,9	73,5	75,8	74,9	73,8	74,9	73,9	75,5	74,9
	\varnothing	75,9	74,9	73,3	75,7	75,2	73,9	75,5	74,2	75,5	74,8
	Max	82,8	82,5	79,8	83,2	81,9	82,5	82,4	81,9	81,3	79,6
$\{\alpha, t_b, T_\psi, T_\delta\}$	Min	92,0	94,6	95,6	92,3	94,9	95,0	92,9	90,3	88,7	93,5
	Med	79,6	82,9	81,5	80,9	80,5	79,8	81,1	83,9	79,5	81,5
	\varnothing	79,6	82,5	81,0	80,6	80,8	79,4	83,0	83,7	81,0	82,3
	Max	85,9	87,1	85,8	84,4	84,2	85,3	86,9	91,9	84,2	87,9
$\{\alpha, t_b, T_\theta, T_\delta\}$	Min	90,8	92,9	96,3	94,6	91,8	94,7	92,7	90,8	89,6	91,8
	Med	79,8	82,8	81,1	80,9	81,0	79,5	82,8	83,7	80,6	82,1
	\varnothing	84,7	87,1	84,7	84,8	85,7	85,6	86,6	92,9	83,5	88,0
	Max	108,3	107,6	102,1	99,8	99,8	103,5	105,1	104,1	96,5	101,0
$\{\alpha, t_b, T_\psi, T_\theta, T_\delta\}$	Min	110,8	113,0	113,6	114,3	115,5	115,4	113,6	110,0	109,6	112,4
	Med	76,4	78,0	77,8	78,3	78,0	76,8	77,7	77,2	78,1	76,3
	\varnothing	95,4	93,1	87,6	90,4	91,0	90,4	93,5	95,9	87,4	91,6
	Max	120,1	122,5	115,7	117,8	123,1	124,1	117,7	118,4	112,6	110,8

E. Ergebnisse der Detektion

Die Tabelle E.27 zeigt die Ergebnisse der Datensätze #6 – #10.

Tabelle E.27.: Ergebnisübersicht (2) der Segmentierung

		#6		#7		#8		#9		#10	
		c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I
$\{\alpha, t_b, T_\psi\}$	Min	117,9	118,7	125,4	124,2	127,1	123,4	123,3	122,1	122,5	123,6
	Med	103,3	100,3	106,5	105,4	109,6	109,0	106,9	103,4	100,5	102,4
	\emptyset	103,8	100,4	109,1	106,2	110,8	111,1	109,0	102,9	101,7	102,1
	Max	106,2	98,5	111,6	110,1	109,7	112,6	109,8	108,6	103,2	106,0
$\{\alpha, t_b, T_\theta\}$	Min	96,4	97,3	98,6	95,1	99,8	99,4	99,0	98,4	98,4	99,5
	Med	74,7	74,6	76,4	76,6	77,9	77,5	76,9	75,9	75,5	75,2
	\emptyset	74,7	74,3	75,7	76,9	77,5	77,3	76,4	77,2	75,2	74,9
	Max	86,4	84,0	88,8	86,2	89,7	85,5	89,3	87,6	81,3	84,5
$\{\alpha, t_b, T_\delta\}$	Min	79,9	81,4	80,7	83,7	83,3	87,5	85,7	80,3	82,5	80,5
	Med	78,4	79,9	81,0	81,1	79,9	85,8	84,7	79,5	78,9	79,2
	\emptyset	77,9	79,6	81,0	80,8	80,0	85,9	84,7	80,2	78,9	79,2
	Max	79,9	81,8	81,8	83,2	82,1	87,8	85,7	80,9	78,8	82,8
$\{\alpha, t_b, T_\psi, T_\theta\}$	Min	94,0	94,4	96,1	94,9	98,2	98,0	96,7	96,8	95,2	95,5
	Med	75,6	75,0	74,4	76,8	74,8	77,0	77,2	75,8	74,9	76,6
	\emptyset	75,5	74,9	74,4	76,7	74,7	76,9	77,3	76,3	74,8	76,4
	Max	83,1	81,0	81,6	82,2	83,3	83,9	86,8	83,5	78,7	82,7
$\{\alpha, t_b, T_\psi, T_\delta\}$	Min	87,7	90,0	91,1	90,5	94,7	95,4	93,0	91,2	92,0	90,5
	Med	78,3	80,7	81,1	81,8	79,8	87,1	83,8	80,2	78,8	80,8
	\emptyset	79,3	81,5	81,8	82,9	80,6	87,5	85,4	80,7	79,1	82,1
	Max	84,1	84,3	88,1	85,8	86,2	91,0	89,5	83,6	82,1	86,5
$\{\alpha, t_b, T_\theta, T_\delta\}$	Min	88,7	91,2	92,6	92,4	95,2	93,6	93,3	91,2	91,5	91,5
	Med	78,4	81,2	81,8	82,3	80,6	87,4	85,4	80,2	78,4	81,1
	\emptyset	84,4	83,1	88,8	84,8	84,7	91,9	89,0	84,1	81,5	85,6
	Max	98,9	91,6	107,4	103,0	103,6	109,9	104,7	105,5	97,2	97,5
$\{\alpha, t_b, T_\psi, T_\theta, T_\delta\}$	Min	108,2	109,3	114,3	109,6	114,5	112,0	109,4	113,0	113,9	112,1
	Med	77,5	77,0	78,4	79,0	76,5	80,6	79,9	75,9	75,9	77,9
	\emptyset	90,8	84,7	95,8	91,3	90,6	98,9	94,7	91,1	87,3	88,8
	Max	118,9	106,0	122,3	119,3	121,4	120,4	123,0	119,3	114,2	113,6

Und in Tabelle E.28 sind die Ergebnisse der Datensätze #11 – #15 aufgeführt.

Tabelle E.28.: Ergebnisübersicht (3) der Segmentierung

		#11		#12		#13		#14		#15	
		c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I	c_I	\hat{c}_I
$\{\alpha, t_b, T_\psi\}$	Min	118,4	122,5	125,7	121,9	123,8	120,5	123,1	123,9	122,3	128,8
	Med	98,4	104,0	108,8	105,2	107,3	101,7	107,4	104,8	104,6	110,8
	\emptyset	101,6	104,6	109,3	105,0	107,4	103,4	109,2	105,5	104,8	110,9
	Max	107,2	104,8	112,0	110,4	109,8	107,4	110,8	108,5	105,7	110,5
$\{\alpha, t_b, T_\theta\}$	Min	97,2	98,1	98,3	97,7	97,6	95,1	101,1	98,1	96,5	100,1
	Med	75,3	75,5	73,2	78,9	77,5	74,0	75,7	74,5	75,5	76,0
	\emptyset	74,7	75,6	72,8	78,6	77,2	74,1	75,5	73,9	75,3	75,7
	Max	83,9	86,3	81,3	84,9	87,6	86,6	84,7	81,9	84,2	87,3
$\{\alpha, t_b, T_\delta\}$	Min	77,4	84,5	79,9	81,0	83,2	76,3	83,0	80,5	83,3	85,2
	Med	77,2	82,6	76,6	80,9	83,2	75,3	82,2	79,5	82,7	82,6
	\emptyset	77,1	82,6	76,4	80,9	83,0	75,5	82,4	79,7	83,1	82,5
	Max	80,4	81,4	75,6	82,4	84,0	79,5	83,7	79,7	86,3	83,1
$\{\alpha, t_b, T_\psi, T_\theta\}$	Min	96,6	95,7	95,4	96,3	97,5	94,3	98,0	97,3	94,5	96,9
	Med	74,6	75,8	72,6	76,6	76,3	73,5	73,9	73,0	74,9	75,1
	\emptyset	74,8	75,8	72,6	76,9	76,1	73,5	73,7	73,0	74,7	75,1
	Max	81,5	82,0	78,7	82,5	84,6	83,0	81,5	78,4	81,3	84,3
$\{\alpha, t_b, T_\psi, T_\delta\}$	Min	87,9	92,2	92,2	90,8	94,2	90,9	94,1	90,7	92,0	96,0
	Med	78,5	81,7	76,1	81,3	82,6	77,1	82,8	78,5	82,9	82,8
	\emptyset	80,0	81,4	75,8	81,1	83,9	78,5	83,7	79,8	84,8	83,3
	Max	83,2	82,4	81,2	85,1	88,2	83,2	89,9	85,0	89,0	86,9
$\{\alpha, t_b, T_\theta, T_\delta\}$	Min	91,1	91,3	91,2	89,5	93,9	89,6	94,0	92,5	91,6	94,1
	Med	79,1	81,5	75,7	81,1	83,1	77,4	83,5	79,3	84,6	83,7
	\emptyset	82,3	82,8	82,1	84,3	87,4	83,3	89,6	84,7	88,5	87,5
	Max	106,6	102,7	106,4	100,9	107,6	101,2	102,0	102,3	106,2	104,0
$\{\alpha, t_b, T_\psi, T_\theta, T_\delta\}$	Min	108,7	111,4	113,8	109,9	113,6	110,2	114,0	112,1	112,8	115,9
	Med	74,9	78,7	75,0	77,8	78,5	74,8	77,5	75,8	77,7	77,4
	\emptyset	89,6	92,6	91,1	88,3	96,6	88,8	95,1	89,9	94,4	93,8
	Max	119,5	118,1	118,7	121,3	121,7	119,1	119,3	118,8	116,3	118,0

E.2.2. Parameterwahl II

Die Tabellen E.29-E.30 listen die Ergebnisse der Kostenfunktionen c_{II} bzw. \hat{c}_{II} (Gleichung 3.112 aus Kapitel 3.7.5) auf, welche mit dem Minimum, dem Median, dem arithmetischen Mittel \varnothing und dem Maximum der 15 einzeln optimierten Parametersätze (siehe Kapitel 3.7.5) für die Lern- bzw. Testdatensätze mit der Detektion aus Kapitel 3.4.4 nach der Segmentierung aus Kapitel 3.4.3 (mit den Parametern aus Kapitel 4.4.3) bestimmt wurden. Die entsprechenden Parameter sind in Tabelle 4.15 im Kapitel 4.4.3 aufgeführt. Für jeden Datensatz wurden jeweils die zwei besten Ergebnisse farblich markiert (bestes Ergebnis c_{II} in Blau, zweitbestes Ergebnis c_{II} in Cyan, bestes Ergebnis \hat{c}_{II} in Rot und das zweitbeste Ergebnis \hat{c}_{II} in Magenta). Die Anzahl $N_{c_{II}}$ bzw. $N_{\hat{c}_{II}}$ in der Tabelle 4.15 aus Kapitel 4.4.3 entsprechen der Anzahl der farblich hervorgehobenen Datensätze in den Tabellen E.29-E.31 (beste und zweitbeste Ergebnisse).

Tabelle E.29 zeigt die Ergebnisse der ersten fünf Datensätze.

Tabelle E.29.: Ergebnisübersicht (1) der Detektion nach der Segmentierung
Parametersatz (4) der Segmentierung: $\{T_\psi, T_\theta\}$

		#1		#2		#3		#4		#5	
		$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
$\{b_\psi\}$	Min	9,5	10,7	10,1	11,9	10,6	10,9	9,7	9,1	9,7	10,3
	Med	7,4	8,9	7,4	8,4	7,5	7,6	8,2	7,2	8,1	8,5
	\varnothing	7,6	8,9	7,8	8,5	7,8	7,5	8,5	7,7	8,5	8,6
	Max	7,5	9,4	7,8	8,7	7,9	7,9	8,3	7,2	8,4	9,0
$\{b_\theta\}$	Min	5,4	6,7	4,6	5,5	5,1	5,3	4,9	4,9	4,2	5,3
	Med	5,6	6,7	4,4	5,7	5,1	5,1	5,1	5,2	4,8	5,7
	\varnothing	5,7	6,8	4,3	5,6	5,2	5,0	4,9	5,0	4,6	5,8
	Max	7,4	8,6	5,8	7,3	5,9	5,6	6,3	5,5	4,7	5,3
$\{b_\delta\}$	Min	19,5	20,3	16,6	22,5	17,8	20,0	19,2	19,4	21,3	20,4
	Med	20,3	21,9	18,0	23,7	18,9	21,0	20,6	20,2	22,9	22,6
	\varnothing	19,8	21,5	17,5	23,1	18,7	21,0	20,2	19,9	22,5	22,0
	Max	18,0	20,6	16,0	21,2	16,0	19,3	17,6	17,8	19,5	19,6
$\{b_\psi, b_\theta\}$	Min	2,0	2,9	0,8	2,1	1,6	2,2	1,5	1,2	1,0	1,4
	Med	1,8	2,8	0,8	2,1	1,5	2,0	1,4	1,0	1,0	1,3
	\varnothing	1,8	2,6	0,7	1,9	1,6	1,8	1,4	1,2	1,0	1,4
	Max	1,4	2,4	0,7	1,4	1,3	1,3	1,1	1,0	1,0	1,5
$\{b_\psi, b_\delta\}$	Min	2,7	3,0	1,2	3,3	2,7	2,9	2,1	1,1	1,5	2,2
	Med	2,7	3,5	1,3	2,9	2,2	2,5	2,3	1,6	1,3	1,8
	\varnothing	2,7	3,6	1,3	3,0	2,3	2,6	2,3	1,6	1,2	1,9
	Max	2,7	3,4	1,5	3,0	2,2	2,7	2,3	1,8	1,5	1,9
$\{b_\theta, b_\delta\}$	Min	2,1	3,0	1,4	2,5	1,7	1,8	1,4	1,3	0,8	1,5
	Med	2,0	3,1	1,1	1,9	1,6	1,7	1,3	1,1	1,1	1,3
	\varnothing	1,9	3,1	1,0	1,9	1,6	1,7	1,3	1,1	1,1	1,3
	Max	2,1	2,9	1,1	2,0	1,9	1,6	1,3	1,0	1,1	1,3

Die Tabelle E.30 zeigt die Ergebnisse der Datensätze #6 – #10.

Tabelle E.30.: Ergebnisübersicht (2) der Detektion nach der Segmentierung
Parametersatz (4) der Segmentierung: $\{T_\psi, T_\theta\}$

		#6		#7		#8		#9		#10	
		$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
$\{b_\psi\}$	Min	10,1	11,5	10,0	9,8	11,0	12,1	11,4	12,3	9,3	11,9
	Med	8,3	8,7	8,3	6,9	7,8	9,8	8,1	8,1	6,1	8,8
	\emptyset	8,6	9,4	8,8	7,1	8,2	10,2	8,6	8,3	6,4	9,3
	Max	8,5	8,7	8,4	7,2	7,8	9,9	8,2	8,2	6,3	8,6
$\{b_\theta\}$	Min	5,1	5,1	5,6	6,0	5,5	5,5	5,6	6,1	5,6	5,9
	Med	5,4	5,2	6,1	6,3	6,4	5,5	5,5	6,2	5,5	6,4
	\emptyset	5,3	5,2	5,9	6,1	6,4	5,6	5,4	6,1	5,4	6,2
	Max	5,2	5,6	4,2	6,5	6,3	6,2	7,3	5,7	5,4	7,4
$\{b_\delta\}$	Min	22,6	21,4	20,7	21,0	18,2	17,7	21,3	19,4	20,0	21,1
	Med	23,8	22,6	21,9	22,7	19,7	19,1	22,8	20,7	21,3	21,5
	\emptyset	23,6	22,5	21,3	21,8	19,6	18,9	22,6	20,6	21,1	21,5
	Max	21,7	19,3	19,4	19,8	17,8	17,2	20,6	17,9	17,0	18,7
$\{b_\psi, b_\theta\}$	Min	2,1	2,4	2,6	2,1	2,3	2,7	2,9	2,7	1,3	3,1
	Med	2,0	2,3	2,5	1,8	2,4	2,6	2,7	2,5	1,4	2,8
	\emptyset	1,9	2,2	2,4	2,0	2,2	2,5	2,5	2,4	1,5	2,8
	Max	1,5	1,7	2,2	1,4	1,9	2,0	2,1	1,9	1,4	2,3
$\{b_\psi, b_\delta\}$	Min	2,3	3,1	3,1	3,1	1,8	3,5	3,3	2,5	2,0	3,4
	Med	2,4	2,5	3,0	2,9	2,1	3,0	3,2	2,9	1,9	3,0
	\emptyset	2,5	2,6	3,0	2,9	2,2	3,0	3,3	2,9	1,9	3,1
	Max	2,8	2,7	3,0	2,7	2,3	3,1	3,2	2,9	1,9	2,9
$\{b_\theta, b_\delta\}$	Min	2,0	2,0	2,6	2,1	2,4	2,9	2,5	2,0	2,2	2,3
	Med	1,9	1,9	2,4	1,5	2,2	2,2	2,3	1,9	2,1	2,1
	\emptyset	2,0	1,9	2,3	1,6	2,1	2,2	2,3	1,9	2,0	2,1
	Max	2,0	2,0	2,5	1,5	2,7	2,2	3,1	1,9	2,3	2,4

E. Ergebnisse der Detektion

Und in Tabelle E.31 sind die Ergebnisse der Datensätze #11 – #15 aufgeführt.

Tabelle E.31.: Ergebnisübersicht (3) der Detektion nach der Segmentierung
Parametersatz (4) der Segmentierung: $\{T_\psi, T_\theta\}$

		#11		#12		#13		#14		#15	
		$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$	$c_{II}[\%]$	$\hat{c}_{II}[\%]$
$\{b_\psi\}$	Min	10,4	10,7	9,9	10,3	10,3	11,0	10,3	10,4	10,5	10,6
	Med	8,5	8,1	8,4	7,7	8,1	8,5	8,4	8,6	8,9	8,5
	\emptyset	8,7	8,7	8,4	7,8	8,4	9,1	8,6	9,3	9,1	8,6
	Max	8,8	8,0	8,9	8,0	8,6	8,6	8,7	8,9	9,4	9,0
$\{b_\theta\}$	Min	4,7	5,5	4,5	5,7	5,1	5,1	5,8	4,5	6,0	6,2
	Med	4,8	5,1	5,0	5,8	5,4	4,9	5,7	4,6	5,4	6,0
	\emptyset	4,8	5,0	4,9	5,6	5,5	5,0	5,8	4,6	5,3	6,0
	Max	6,1	4,4	5,5	5,4	7,1	3,8	7,1	5,2	5,1	7,1
$\{b_\delta\}$	Min	19,6	21,8	18,8	20,6	19,2	19,2	16,2	16,7	18,7	16,6
	Med	20,7	23,3	19,8	22,2	21,1	20,1	18,1	18,2	20,5	19,0
	\emptyset	20,6	22,8	19,9	21,6	20,6	20,0	17,7	18,0	20,0	18,5
	Max	17,8	19,0	17,6	18,0	18,4	16,8	13,8	15,8	19,2	16,7
$\{b_\psi, b_\theta\}$	Min	1,7	2,0	1,1	2,2	1,2	2,1	1,6	1,2	1,2	2,0
	Med	1,4	2,0	1,0	2,0	1,1	1,9	1,5	1,1	1,0	1,8
	\emptyset	1,5	1,8	1,0	2,0	1,3	1,9	1,6	1,2	1,1	1,7
	Max	1,2	1,3	0,9	1,9	1,1	1,4	1,3	0,9	0,8	1,5
$\{b_\psi, b_\delta\}$	Min	2,1	2,8	1,4	2,4	2,1	2,2	2,3	1,6	2,3	1,9
	Med	1,7	2,6	1,2	2,5	1,6	2,3	2,1	1,8	1,9	2,0
	\emptyset	1,7	2,7	1,3	2,5	1,7	2,3	2,1	1,7	1,9	2,0
	Max	2,1	2,5	1,2	2,3	1,7	2,4	2,3	1,9	1,7	2,0
$\{b_\theta, b_\delta\}$	Min	1,4	2,0	1,3	2,5	1,4	2,4	1,2	1,4	1,6	1,6
	Med	1,0	1,5	1,0	2,3	1,1	2,1	1,2	1,1	0,9	1,2
	\emptyset	1,0	1,5	1,0	2,4	1,1	2,1	1,3	1,1	0,9	1,3
	Max	1,3	1,9	1,0	2,2	1,5	2,0	1,3	1,3	1,1	1,2

F. Ergebnisse zur Parameterwahl der Posenbestimmung

Das Vorgehen zum Optimieren und Bestimmen der Parameter zur Posenbestimmung aus Kapitel 3.5.1 bzw. 3.5.2 ist in Kapitel 3.7.6 beschrieben. Die Posenbestimmung ist in zwei Schritte (A und B) eingeteilt: A) die *Grobe Posenbestimmung* aus Kapitel 3.5.1 und B) die *Posenverfeinerung* aus Kapitel 3.5.2. Für jeden dieser Schritte (A und B) wurden optimale Parameter nach der *Parameteroptimierung A* (Kapitel 3.7.6) bzw. *Parameteroptimierung B* (Kapitel 3.7.6) pro Lerndatensatz bestimmt, die im Kapitel 4.5.1 in den Tabellen 4.17 bzw. 4.18 aufgeführt sind. Um eine *Parameterwahl* treffen zu können wurden in Kapitel 4.5.2 je Schritt (A bzw. B) und Parameter das Minimum, der Median, das arithmetische Mittel \emptyset und das Maximum der pro Lerndatensatz optimierten Parameter (Tabelle 4.17 bzw. 4.18) bestimmt. Diese sind in den Tabellen 4.19 bzw. 4.20 im Kapitel 4.5.2 zu finden.

Für alle Kombinationen der Parametersätze aus Tabelle 4.19 (*Parameteroptimierung A*) und Tabelle 4.20 (*Parameteroptimierung B*) wurde die Kostenfunktion c_B bzw. \hat{c}_B nach Gleichung 3.122 (Kapitel 3.7.6) mit den Lern- bzw. Testdatensätzen (siehe Kapitel 3.7.5) berechnet. Diese Ergebnisse (c_B bzw. \hat{c}_B) sind für die 15 Lern- bzw. Testdatensätze in den Tabellen F.1-F.5 eingetragen.

Neben den in Metern angegebenen Kosten c_B bzw. \hat{c}_B der Gleichung 3.122 (Kapitel 3.7.6), sind die durchschnittlichen Kosten $c_{\emptyset B}$ bzw. $\hat{c}_{\emptyset B}$ pro berücksichtigtem Frame aus den Lern- bzw. Testdatensätzen in Zentimetern in den Tabellen F.1-F.5 angegeben. Auf die durchschnittlichen Kosten $c_{\emptyset B}$ bzw. $\hat{c}_{\emptyset B}$ haben nur die berücksichtigten Frames der Lern- bzw. Testdatensätze (mit je 199 Frames pro Datensatz) einen Einfluss. So wurden ca. 1 – 5 % der Daten eines Lern- bzw. Testdatensatzes nicht berücksichtigt ($\overline{d_B} = 0$ in Gleichung 3.122), da keine bzw. eine unrealistische Detektion festgestellt werden konnte (siehe *Parameteroptimierung B* bzw. A aus Kapitel 3.7.6). $c_{\emptyset B}$ und $\hat{c}_{\emptyset B}$ sind lediglich zusätzliche Angaben, die den Fehler entsprechend der Kostenfunktion nach Gleichung 3.122 (Kapitel 3.7.6) auf die verwendeten Frames beschreiben.

Aus den aufgeführten Kombinationen der Parameter der beiden Schritte (A und B, erste und zweite Spalte der Tabellen F.1-F.5) wurden für jeden Datensatz die jeweils zwei besten Ergebnisse farblich markiert (bestes Ergebnis c_B in Blau, zweitbestes Ergebnis c_B in Cyan, bestes Ergebnis \hat{c}_B in Rot und das zweitbeste Ergebnis \hat{c}_B in Magenta).

F. Ergebnisse zur Parameterwahl der Posenbestimmung

Tabelle F.1 zeigt die Ergebnisse der ersten drei Datensätze und Tabelle F.2, die der Datensätze #4 – #6.

Tabelle F.1.: Ergebnisübersicht (1) der Posenbestimmung aus Kapitel 3.5.1 und 3.5.2
Datensätze #1 - #3

A	B	Einheiten: c_B bzw. \hat{c}_B in m und $c_{\emptyset B}$ bzw. $c_{\hat{\emptyset} B}$ in cm											
		#1				#2				#3			
		c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$	c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$	c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$
Min	Min	7,9	4,1	8,5	4,4	8,3	4,2	7,6	4,0	9,0	4,7	8,6	4,5
Min	Med	7,3	3,8	8,1	4,2	8,1	4,1	7,4	3,9	8,6	4,4	8,0	4,1
Min	\emptyset	7,4	3,9	8,1	4,2	8,3	4,2	7,5	4,0	8,6	4,4	8,1	4,2
Min	Max	8,3	4,4	8,7	4,6	9,3	4,7	7,9	4,2	9,4	4,8	8,8	4,6
Med	Min	7,6	4,0	8,2	4,3	8,2	4,1	7,7	4,1	8,3	4,3	8,7	4,5
Med	Med	6,9	3,6	7,6	4,0	7,7	3,9	7,6	4,0	7,6	3,9	7,9	4,1
Med	\emptyset	7,1	3,7	7,5	3,9	8,0	4,0	7,5	3,9	7,9	4,1	8,0	4,2
Med	Max	8,0	4,2	8,2	4,3	9,1	4,6	7,9	4,2	8,6	4,4	8,6	4,5
\emptyset	Min	7,8	4,1	7,8	4,1	8,5	4,3	7,5	3,9	8,4	4,4	8,6	4,5
\emptyset	Med	7,1	3,7	7,5	3,9	8,1	4,1	7,5	3,9	7,7	4,0	7,7	4,0
\emptyset	\emptyset	7,3	3,8	7,3	3,8	8,3	4,2	7,4	3,9	8,0	4,1	7,9	4,1
\emptyset	Max	8,2	4,3	8,1	4,2	9,5	4,8	7,9	4,2	8,8	4,5	8,4	4,4
Max	Min	7,7	4,0	6,7	3,5	7,4	3,8	6,8	3,6	8,0	4,1	7,8	4,0
Max	Med	6,8	3,6	6,7	3,5	7,0	3,5	6,7	3,5	6,9	3,6	7,2	3,7
Max	\emptyset	7,0	3,6	6,6	3,4	7,2	3,7	6,6	3,5	7,4	3,8	7,3	3,8
Max	Max	7,8	4,1	7,1	3,7	8,4	4,3	7,2	3,8	7,8	4,1	7,9	4,1

Tabelle F.2.: Ergebnisübersicht (2) der Posenbestimmung aus Kapitel 3.5.1 und 3.5.2
Datensätze #4 - #6

A	B	Einheiten: c_B bzw. \hat{c}_B in m und $c_{\emptyset B}$ bzw. $c_{\hat{\emptyset} B}$ in cm											
		#4				#5				#6			
		c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$	c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$	c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$
Min	Min	8,9	4,6	8,2	4,2	8,4	4,3	8,5	4,5	8,5	4,4	8,3	4,3
Min	Med	7,9	4,1	7,5	3,9	8,3	4,3	8,1	4,3	7,8	4,1	8,1	4,2
Min	\emptyset	8,1	4,1	7,7	3,9	8,3	4,3	8,4	4,4	8,0	4,1	8,3	4,3
Min	Max	9,3	4,8	8,3	4,3	8,6	4,4	8,6	4,5	8,8	4,6	9,0	4,7
Med	Min	7,9	4,0	8,1	4,2	7,6	3,9	9,2	4,8	8,5	4,4	7,9	4,1
Med	Med	7,5	3,8	7,4	3,8	7,3	3,8	8,8	4,6	7,7	4,0	7,5	3,9
Med	\emptyset	7,6	3,9	7,4	3,8	7,6	3,9	8,8	4,6	7,8	4,1	7,7	4,0
Med	Max	8,6	4,4	8,2	4,2	7,8	4,0	9,2	4,8	8,5	4,4	8,5	4,4
\emptyset	Min	7,8	4,0	7,9	4,1	7,4	3,8	8,8	4,6	8,2	4,2	7,9	4,1
\emptyset	Med	7,4	3,8	7,3	3,7	7,3	3,8	8,4	4,4	7,3	3,8	7,5	3,9
\emptyset	\emptyset	7,7	3,9	7,2	3,7	7,5	3,9	8,4	4,4	7,5	3,9	7,7	4,0
\emptyset	Max	8,8	4,5	8,1	4,2	7,9	4,1	8,8	4,6	8,3	4,3	8,6	4,4
Max	Min	8,1	4,2	7,3	3,8	7,2	3,7	7,7	4,0	7,8	4,1	7,6	3,9
Max	Med	7,4	3,8	6,9	3,6	7,0	3,6	7,2	3,8	6,7	3,5	7,4	3,8
Max	\emptyset	7,5	3,9	7,0	3,6	7,1	3,6	7,0	3,7	6,8	3,6	7,7	4,0
Max	Max	8,6	4,4	8,0	4,1	7,5	3,9	7,5	3,9	7,6	4,0	8,6	4,4

In Tabelle F.3 sind die Ergebnisse der Datensätze #7 – #9 gezeigt.

Tabelle F.3.: Ergebnisübersicht (3) der Posenbestimmung aus Kapitel 3.5.1 und 3.5.2

Einheiten: c_B bzw. \hat{c}_B in m und $c_{\emptyset B}$ bzw. $c_{\hat{\emptyset} B}$ in cm													
A	B	#7				#8				#9			
		c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$	c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$	c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$
Min	Min	8,0	4,1	8,0	4,1	9,0	4,7	9,1	4,7	8,8	4,6	9,1	4,8
Min	Med	7,6	3,9	7,9	4,1	8,8	4,6	8,4	4,3	8,4	4,4	8,5	4,4
Min	\emptyset	7,3	3,8	8,0	4,1	8,8	4,6	8,2	4,2	9,0	4,7	8,6	4,5
Min	Max	8,2	4,2	8,8	4,6	9,8	5,1	9,0	4,7	9,7	5,1	9,2	4,8
Med	Min	7,9	4,1	7,7	4,0	8,2	4,3	8,8	4,6	7,9	4,1	8,7	4,6
Med	Med	7,8	4,0	7,2	3,7	8,2	4,3	8,2	4,2	7,6	4,0	8,2	4,3
Med	\emptyset	7,3	3,8	7,7	4,0	8,0	4,2	8,1	4,2	8,1	4,2	8,3	4,4
Med	Max	8,1	4,2	8,4	4,4	9,0	4,7	8,9	4,6	8,9	4,6	8,9	4,7
\emptyset	Min	7,8	4,0	8,0	4,1	7,8	4,1	8,9	4,6	7,7	4,0	8,8	4,6
\emptyset	Med	7,6	3,9	7,7	4,0	8,0	4,2	8,4	4,3	7,3	3,8	8,0	4,2
\emptyset	\emptyset	7,2	3,7	8,0	4,1	7,9	4,1	8,3	4,3	7,9	4,1	8,2	4,3
\emptyset	Max	8,0	4,1	8,8	4,6	8,8	4,6	9,1	4,7	8,5	4,4	8,8	4,6
Max	Min	7,3	3,8	7,2	3,8	6,7	3,5	7,9	4,1	7,7	4,0	8,0	4,2
Max	Med	7,0	3,6	7,0	3,6	7,0	3,6	7,2	3,7	7,4	3,8	7,1	3,7
Max	\emptyset	7,1	3,7	7,2	3,7	6,9	3,6	7,3	3,7	7,8	4,0	7,1	3,7
Max	Max	7,6	3,9	7,7	4,0	8,0	4,2	7,9	4,1	8,8	4,6	7,9	4,1

Die Tabelle F.4 zeigt die Ergebnisse der Datensätze #10 – #12.

Tabelle F.4.: Ergebnisübersicht (4) der Posenbestimmung aus Kapitel 3.5.1 und 3.5.2

Einheiten: c_B bzw. \hat{c}_B in m und $c_{\emptyset B}$ bzw. $c_{\hat{\emptyset} B}$ in cm													
A	B	#10				#11				#12			
		c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$	c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$	c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\hat{\emptyset} B}$
Min	Min	8,0	4,1	8,3	4,3	10,1	5,2	8,2	4,3	8,4	4,3	8,0	4,3
Min	Med	8,0	4,1	7,5	3,9	10,0	5,1	7,8	4,0	7,6	3,9	7,5	4,0
Min	\emptyset	8,2	4,2	7,7	4,0	9,9	5,1	7,9	4,1	7,5	3,8	7,5	4,0
Min	Max	8,4	4,4	8,4	4,4	10,1	5,2	8,9	4,6	8,2	4,2	8,3	4,4
Med	Min	7,1	3,7	8,3	4,4	8,3	4,2	8,4	4,4	8,0	4,1	7,9	4,2
Med	Med	7,0	3,6	7,7	4,0	8,1	4,2	8,0	4,2	7,2	3,7	7,9	4,2
Med	\emptyset	7,2	3,7	7,6	4,0	7,8	4,0	8,1	4,2	7,3	3,7	7,7	4,1
Med	Max	7,5	3,9	8,7	4,6	8,6	4,4	9,0	4,6	7,9	4,0	8,4	4,5
\emptyset	Min	7,1	3,7	8,3	4,3	8,6	4,4	7,9	4,1	7,8	4,0	7,7	4,1
\emptyset	Med	7,1	3,7	7,5	3,9	8,3	4,2	7,5	3,9	7,1	3,6	7,7	4,1
\emptyset	\emptyset	7,2	3,7	7,5	3,9	8,1	4,2	7,6	4,0	7,2	3,7	7,3	3,9
\emptyset	Max	7,5	3,9	8,5	4,5	8,7	4,5	8,4	4,4	7,7	3,9	8,2	4,3
Max	Min	7,2	3,7	7,6	4,0	8,2	4,2	7,7	4,0	7,9	4,0	7,2	3,8
Max	Med	6,8	3,5	6,8	3,6	8,1	4,1	7,0	3,6	7,0	3,5	7,1	3,8
Max	\emptyset	7,1	3,7	6,7	3,5	7,7	3,9	7,1	3,7	7,0	3,6	7,0	3,7
Max	Max	7,5	3,9	7,7	4,0	8,3	4,3	8,1	4,2	7,5	3,8	7,5	4,0

F. Ergebnisse zur Parameterwahl der Posenbestimmung

Und in Tabelle F.5 sind die Ergebnisse der Datensätze #13 – #15 aufgeführt.

Tabelle F.5.: Ergebnisübersicht (5) der Posenbestimmung aus Kapitel 3.5.1 und 3.5.2

Einheiten: c_B bzw. \hat{c}_B in m und $c_{\emptyset B}$ bzw. $c_{\emptyset \hat{B}}$ in cm													
A	B	#13				#14				#15			
		c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\emptyset \hat{B}}$	c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\emptyset \hat{B}}$	c_B	$c_{\emptyset B}$	\hat{c}_B	$c_{\emptyset \hat{B}}$
Min	Min	8,2	4,3	8,0	4,1	8,3	4,3	8,5	4,3	8,6	4,4	8,6	4,4
Min	Med	7,1	3,7	7,7	3,9	7,7	3,9	7,6	3,9	7,6	3,9	7,9	4,0
Min	\emptyset	7,3	3,8	7,5	3,8	8,0	4,1	8,0	4,1	7,8	4,0	8,2	4,2
Min	Max	8,1	4,2	8,4	4,3	8,7	4,5	8,9	4,5	8,9	4,6	8,7	4,5
Med	Min	8,3	4,3	7,6	3,9	8,6	4,4	8,0	4,1	8,3	4,3	8,3	4,3
Med	Med	7,6	4,0	7,8	4,0	7,5	3,9	7,2	3,7	7,3	3,8	7,7	3,9
Med	\emptyset	7,7	4,0	7,7	3,9	7,7	4,0	7,6	3,9	7,3	3,8	7,8	4,0
Med	Max	8,4	4,3	8,3	4,2	8,5	4,4	8,5	4,3	8,4	4,3	8,6	4,4
\emptyset	Min	8,4	4,3	7,7	4,0	8,0	4,1	8,0	4,1	7,8	4,0	8,2	4,2
\emptyset	Med	7,6	4,0	7,8	4,0	7,4	3,8	7,4	3,8	7,0	3,6	7,5	3,8
\emptyset	\emptyset	7,6	4,0	7,6	3,9	7,5	3,8	7,7	4,0	7,2	3,7	7,6	3,9
\emptyset	Max	8,4	4,4	8,4	4,3	8,4	4,3	8,7	4,4	8,2	4,2	8,4	4,3
Max	Min	7,6	3,9	7,2	3,7	7,6	3,9	7,6	3,9	8,0	4,1	7,6	3,9
Max	Med	7,2	3,8	6,9	3,6	6,8	3,5	6,8	3,5	6,7	3,5	6,8	3,5
Max	\emptyset	7,3	3,8	7,0	3,6	7,1	3,6	7,2	3,6	7,1	3,7	6,9	3,6
Max	Max	7,9	4,1	7,8	4,0	7,7	4,0	8,0	4,1	8,1	4,2	7,7	4,0

Die Auswertung nach der Anzahl N_{c_B} bzw. $N_{\hat{c}_B}$ der jeweils besten/zweitbesten Ergebnisse über alle Datensätze (Lern- bzw. Testdatensätze) findet sich in der Tabelle 4.21 im Kapitel 4.5.2.

Literaturverzeichnis

- [**Abate u. a. 2007**] ABATE, Andrea F. ; NAPPI, Michele ; RICCIO, Daniel ; SABATINO, Gabriele: 2D and 3D face recognition: A survey. In: *Pattern Recognition Letters (PRL)* 28 (2007), Nr. 14, 1885 – 1906
- [**Abaza u. a. 2013**] ABAZA, Ayman ; ROSS, Arun ; HEBERT, Christina ; HARRISON, Mary Ann F. ; NIXON, Mark S.: A survey on ear biometrics. In: *ACM Computing Surveys (CSUR)* 45 (2013), Nr. 2
- [**Abramson 2002**] ABRAMSON, Mark A.: *Pattern search algorithms for mixed variable general constrained optimization problems*. Houston, Texas, Rice University, PhD thesis, August 2002
- [**Adeoye 2010**] ADEOYE, Olufemi S.: A survey of emerging biometric technologies. In: *Computer Applications (IJCA)* 9 (2010), Nr. 10
- [**Afkham u. a. 2008**] AFKHAM, H.M. ; TARGHI, A.T. ; EKLUNDH, J.-O. ; PRONOBIS, A.: Joint visual vocabulary for animal classification. In: *Pattern Recognition (ICPR)* Bd. 19, 2008. – DOI 10.1109/ICPR.2008.4761710. ISSN 1051 – 4651
- [**Ahonen u. a. 2004**] AHONEN, Timo ; HADID, Abdenour ; PIETIKÄINEN, Matti: Face recognition with local binary patterns. In: PAJDLA, Tomás (Hrsg.) ; MATAS, Jiří (Hrsg.): *European Conference on Computer Vision (ECCV)* Bd. 8. Berlin, Heidelberg : Springer, 2004, 469 – 481. – DOI 10.1007/978-3-540-24670-1_36. – ISBN 978-3-5402-1984-2
- [**Ahrendt u. a. 2011**] AHRENDT, Peter ; GREGERSEN, Torben ; KARSTOFT, Henrik: Development of a real-time computer vision system for tracking loose-housed pigs. In: *Computers and Electronics in Agriculture (COMPAG)* 76 (2011), Nr. 2, 169 – 174. – DOI 10.1016/j.compag.2011.01.011. – ISSN 0168-1699
- [**Al-Sharadqah u. Chernov 2009**] AL-SHARADQAH, Ali ; CHERNOV, Nikolai: Error analysis for circle fitting algorithms. In: *Electronic Journal of Statistics (EJS)* 3 (2009), 886 – 911. – DOI 10.1214/09-EJS419
- [**Amidror 2002**] AMIDROR, Isaac: Scattered data interpolation methods for electronic imaging systems: A survey. In: *Electronic Imaging (JEI)* 11 (2002), April, Nr. 2, 157 – 176. – DOI 10.1117/1.1455013
- [**Anderson u. a. 2010**] ANDERSON, Carlos J. R. ; VITORIA, Lobo Niels Da V. ; ROTH, James D. ; M., Waterman J.: A computer aided photo identification

- system based on image pattern recognition with an application to polar bears using whisker spot patterns. In: *Journal of Mammalogy* 91 (2010), Dezember, 1350 – 1359. – DOI 10.1644/09-MAMM-A-425.1. – ISSN 0022-2372
- [**Andreopoulos u. Tsotsos 2013**] ANDREOPOULOS, Alexander ; TSOTSOS, John K.: 50 years of object recognition: Directions forward. In: *Computer Vision and Image Understanding (CVIU)* 117 (2013), Nr. 8, 827 – 891
- [**Anglart 2010**] ANGLART, Dorota: *Automatic estimation of body weight and body condition score in dairy cows using 3D imaging technique*. Dept. of Animal Nutrition and Management, Sveriges lantbruksuniversite, Uppsala: SLU, Dept. of Animal Nutrition and Management, Masterthesis, 2010
- [**Anguzza 2013**] ANGUZZA, Umberto: *A method to develop a computer-vision based system for the automatic dairy cow identification and behaviour detection in free stall barns*, University of Catania, PhD thesis, Februar 2013
- [**Antolovic 2008**] ANTOLOVIC, Danko: Review of the hough transform method, with an implementation of the fast hough variant for line detection / Indiana University, and IBM Corporation. 2008 (TR663). – Technical Report. – Department of Computer Science
- [**Araabi u. a. 2000**] ARAABI, B.N. ; KEHTARNAVAZ, N. ; MCKINNEY, T. ; HILLMAN, G. ; WÜRSIG, B.: A string matching computer-assisted system for dolphin photoidentification. In: *Annals of Biomedical Engineering* 28 (2000), Oktober, Nr. 10, 1269 – 1279. – DOI 10.1114/1.1317532. – ISSN 0090-6964
- [**Ardovini u. a. 2008**] ARDOVINI, A. ; CINQUE, L. ; SANGINETO, E.: Identifying elephant photos by multi-curve matching. In: *Pattern Recognition* 41 (2008), Nr. 6, 1867 – 1877. – DOI 10.1016/j.patcog.2007.11.010. – ISSN 0031 – 3203
- [**Ardovini u. a. 2007**] ARDOVINI, Alessandro ; CINQUE, Luigi ; ROCCA, Francesca ; SANGINETO, Enver: A semi-automatic approach to photo identification of wild elephants. In: MARTÍ, Joan (Hrsg.) ; BENEDÍ, JoséMiguel (Hrsg.) ; MENDONÇA, AnaMaria (Hrsg.) ; SERRAT, Joan (Hrsg.): *Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA)* Bd. 3. Berlin, Heidelberg : Springer, 2007 (LNCS), 225 – 232 – DOI 10.1007/978-3-540-72847-4_30. – ISBN 978-3-5407-2846-7
- [**Aurenhammer 1991**] AURENHAMMER, Franz: Voronoi diagrams - a survey of a fundamental geometric data structure. In: *ACM Computing Surveys (CSUR)* 23 (1991), September, Nr. 3, 345 – 405. – DOI 10.1145/116873.116880. – ISSN 0360-0300
- [**Awad u. a. 2013a**] AWAD, A. I. ; ZAWBAA, H. M. ; MAHMOUD, H. A. ; NABI, E. H. H. A. ; FAYED, R. H. ; HASSANIEN, A. E.: A robust cattle identification scheme using muzzle print images. In: *Computer Science and Information Systems (FedCSIS)*, 2013, 529 – 534.

- [**Awad u. a. 2013b**] AWAD, Ali I. ; HASSANIEN, Aboul E. ; ZAWBAA, Hossam M.: A cattle identification approach using live captured muzzle print images. In: AWAD, Ali I. (Hrsg.) ; HASSANIEN, Aboul E. (Hrsg.) ; BABA, Kensuke (Hrsg.): *Advances in Security of Information and Communication Networks* Bd. 381. Berlin, Heidelberg : Springer, 2013, 143 – 152. – DOI 10.1007/978-3-642-40597-6_12. – ISBN 978-3-6424-0596-9
- [**Aydin u. a. 2010**] AYDIN, A. ; CANGAR, O. ; OZCAN, S. E. ; BAHR, C. ; BERCKMANS, D.: Application of a fully automatic analysis tool to assess the activity of broiler chickens with different gait scores. In: *Computers and Electronics in Agriculture (COMPAG)* 73 (2010), 194 – 199. – DOI 10.1016/j.compag.2010.05.004
- [**Bahr u. a. 2008**] BAHR, Claudia ; LEROY, Toon ; SONG, Xiangyu ; MAERTENS, Willem ; VRANKEN, Erik ; NUFFEL, Annelies van ; VANGEYTE, Jürgen ; SONCK, Bart ; BERCKMANS, Daniel: Automatic detection of lameness in dairy cattle by vision analysis of cow's gait. In: *Agricultural Engineering (AgEng)* (2008), Juni, Nr. 1130871
- [**Banerjee u. Woodard 2012**] BANERJEE, Salil P. ; WOODARD, Damon L.: Biometric authentication and identification using keystroke dynamics: A survey. In: *Pattern Recognition Research (JPRR)* 7 (2012), Nr. 1, 116 – 139
- [**Barber u. a. 1996**] BARBER, C. B. ; DOBKIN, David P. ; HUHDANPAA, Hannu: The quickhull algorithm for convex hulls. In: *ACM Transactions on Mathematical Software (TOMS)* 22 (1996), Nr. 4, 469 – 483
- [**Barron u. a. 2008**] BARRON, U. G. ; CORKERY, G. ; BARRY, B. ; BUTLER, F. ; McDONNELL, K. ; WARD, S.: Assessment of retinal recognition technology as a biometric method for sheep identification. In: *Computers and Electronics in Agriculture (COMPAG)* 60 (2008), Nr. 2, 156 – 166. – DOI 10.1016/j.compag.2007.07.010. – ISSN 0168-1699
- [**Barron u. a. 2009**] BARRON, UG ; BUTLER, F ; McDONNELL, K ; WARD, S u. a.: The end of the identity crisis? Advances in biometric markers for animal identification. In: *Irish Veterinary Journal* 62 (2009), Nr. 3, 204 – 208
- [**Barry u. a. 2008**] BARRY, B. ; CORKERY, G. ; GONZALES-BARRON, U. ; MC DONNELL, K. ; BUTLER, F. ; WARD, S.: A longitudinal study of the effect of time on the matching performance of a retinal recognition system for lambs. In: *Computers and Electronics in Agriculture (COMPAG)* 64 (2008), Dezember, Nr. 2, 202 – 211. – DOI 10.1016/j.compag.2008.05.011. – ISSN 0168-1699
- [**Barry u. a. 2007**] BARRY, B ; GONZALES-BARRON, UA ; McDONNELL, K ; BUTLER, F ; WARD, S: Using muzzle pattern recognition as a biometric approach for cattle identification. In: *American Society of Agricultural and Biological Engineers (ASABE)* 50 (2007), Nr. 3, 1073 – 1080. – DOI 10.13031/2013.23121

- [**Bartoli 2007**] BARTOLI, Adrien: A random sampling strategy for piecewise planar scene segmentation. In: *Computer Vision and Image Understanding (CVIU)* 105 (2007), Nr. 1, 42 – 59. – DOI 10.1016/j.cviu.2006.07.011. – ISSN 1077–3142
- [**Bay u. a. 2008**] BAY, Herbert ; ESS, Andreas ; TUYTELAARS, Tinne ; GOOL, Luc van: Speeded-up robust features (SURF). In: *Computer Vision and Image Understanding (CVIU)* 110 (2008), Nr. 3, 346 – 359. – DOI 10.1016/j.cviu.2007.09.014
- [**Berckmans 2006**] BERCKMANS, Daniel: Automatic on-line monitoring of animals by precision livestock farming. In: *Livestock Production and Society* 287 (2006), 287 – 294. ISBN 978–9–0769–9889–3
- [**Berg u. Forsyth 2006**] BERG, T.L. ; FORSYTH, D.A.: Animals on the web. In: *Computer Vision and Pattern Recognition (CVPR)* Bd. 2, 2006, 1463 – 1470. – DOI 10.1109/CVPR.2006.57. – ISSN 1063–6919
- [**Besl u. McKay 1992**] BESL, Paul J. ; MCKAY, Neil D.: A method for registration of 3-D shapes. In: *Pattern Analysis and Machine Intelligence (TPAMI)* 14 (1992), Nr. 2, 239 – 256
- [**Bhatia u. a. 2010**] BHATIA, Nitin u. a.: Survey of nearest neighbor techniques. In: *Computer Science and Information Security (IJCSIS)* 8 (2010), Nr. 2, 302 – 305
- [**Biswas u. Veloso 2012**] BISWAS, Joydeep ; VELOSO, Manuela: Depth camera based indoor mobile robot localization and navigation. In: *Robotics and Automation (ICRA)*, IEEE, Mai 2012, 1697 – 1702.
- [**Blanz u. Vetter 2003**] BLANZ, V. ; VETTER, T.: Face recognition based on fitting a 3D morphable model. In: *Pattern Analysis and Machine Intelligence (TPAMI)* 25 (2003), September, Nr. 9, 1063 – 1074. – DOI 10.1109/TPAMI.2003.1227983. – ISSN 0162–8828
- [**Blanz 2006**] BLANZ, Volker: Face recognition based on a 3D morphable model. In: *Automatic Face and Gesture Recognition* Bd. 7 IEEE, 2006, 617 – 624.
- [**Bo 2010**] BO, Hua: A novel features design method for cat head detection. In: WANG, FuLee (Hrsg.) ; DENG, Hepu (Hrsg.) ; GAO, Yang (Hrsg.) ; LEI, Jingsheng (Hrsg.): *Artificial Intelligence and Computational Intelligence* Bd. 6319. Berlin, Heidelberg : Springer, 2010, 397 – 405. – DOI 10.1007/978–3–642–16530–6_47. – ISBN 978–3–6421–6529–0
- [**Bolle u. a. 2005**] BOLLE, Ruud M. ; CONNELL, Jonathan H. ; PANKANTI, Sharath ; RATHA, Nalini K. ; SENIOR, Andrew W.: The relation between the ROC curve and the CMC. In: *Automatic Identification Advanced Technologies* Bd. 4, 2005, 15 – 20.

- [**Bowyer u. a. 2006**] BOWYER, Kevin W. ; CHANG, Kyong ; FLYNN, Patrick: A survey of approaches and challenges in 3D and multi-modal 3D+ 2D face recognition. In: *Computer Vision and Image Understanding (CVIU)* 101 (2006), Nr. 1
- [**Bowyer u. a. 2008**] BOWYER, Kevin W. ; HOLLINGSWORTH, Karen ; FLYNN, Patrick J.: Image understanding for iris biometrics: A survey. In: *Computer Vision and Image Understanding (CVIU)* 110 (2008), Nr. 2, 281 – 307. – DOI 10.1016/j.cviu.2007.08.005. – ISSN 1077–3142
- [**Bradski 2000**] BRADSKI, G.: The OpenCV library. In: *Dr. Dobb's Journal of Software Tools* (2000)
- [**Bronštejn u. a. 2012**] BRONŠTEJN, I.N. ; MUSIOL, G. ; MÜHLIG, H. ; SEMEND-JAJEW, K.A.: *Taschenbuch der Mathematik*. Harri Deutsch, 2012 (8 ISBN 9783817120086). – ISBN 978–3–8171–2008–6
- [**Buchecker u. a. 2011**] BUCHECKER, Kirsten ; MAHNKE-PLESKER, Sylvia ; BÖHM, Herwart ; WESTHUES, Franz ; BÖHM, Herwart (Hrsg.): *Optimierung der ökologischen Kartoffelproduktion - Interpretation ausgewählter sensorischer Ergebnisse von Bio-Kartoffeln durch den Vergleich der Werte mittels Boxplots*. Johann Heinrich von Thünen-Institut - Bundesforschungsinstitut für Ländliche Räume, Wald und Fischerei, D-Darmstadt : Böhm, Herwart, 2011 (Sonderheft Nr. 348), 99 – 110. – ISBN 978–3–8657–6074–6
- [**Burger u. Burge 2006**] BURGER, W. ; BURGE, M.J. ; MEDIA, Springer Science & B. (Hrsg.): *Digitale Bildverarbeitung*. Berlin, Heidelberg : Springer, 2006 (X. media. press Series). – ISBN 978–3–5403–0940–6
- [**Burghardt 2008**] BURGHARDT, T.: *Visual animal biometrics: Automatic detection and individual identification by coat pattern*, University of Bristol, Phd thesis, Juni 2008
- [**Burghardt u. Calic 2006a**] BURGHARDT, T. ; CALIC, J.: Analysing animal behaviour in wildlife videos using face detection and tracking. In: *Vision, Image and Signal Processing* 153 (2006), Juni, Nr. 3, 305 – 312. – DOI 10.1049/ip-vis:20050052. – ISSN 1350–245X
- [**Burghardt u. Calic 2006b**] BURGHARDT, T. ; CALIC, J.: Real-time face detection and tracking of animals. In: *Neural Network Applications in Electrical Engineering (NEUREL)* Bd. 8, 2006, 27 – 32. – DOI 10.1109/NEUREL.2006.341167
- [**Burghardt u. a. 2004a**] BURGHARDT, Tilo ; ČALIĆ, Janko ; THOMAS, Barry T.: Tracking animals In wildlife videos using face detection. In: *Integration of Knowledge, Semantics and Digital Media Technology (EWIMT)*, 2004

- [**Burghardt u. Campbell 2007**] BURGHARDT, Tilo ; CAMPBELL, Neill: Individual animal identification using visual biometrics on deformable coat patterns. In: *Computer Vision Systems (ICVS)* Bd. 5, 2007. – DOI 10.2390/biecoll-icvs2007-121
- [**Burghardt u. a. 2004b**] BURGHARDT, Tilo ; THOMAS, Barry ; BARHAM, Peter J. ; CALIC, Janko: Automated visual recognition of individual african penguins. In: *International Penguin Conference (IPC)* 5 (2004), September
- [**Busetti 2003**] BUSETTI, Franco: Simulated annealing overview / JP Morgan, Italy. Citeseer, 2003. – Technical Report
- [**Cai u. Li 2013**] CAI, Cheng ; LI, Jianqiao: Cattle face recognition using local binary pattern descriptor. In: *Signal and Information Processing Association (APSIPA)*, 2013. – DOI 10.1109/APSIPA.2013.6694369
- [**Ćalić u. a. 2005**] ĆALIĆ, J. ; CAMPBELL, N. ; CALWAY, A. ; MIRMEHDI, M. ; BURGHARDT, T. ; HANNUNA, S. ; KONG, C. ; PORTER, S. ; CANAGARAJAH, N. ; BULL, D.: Towards intelligent content based retrieval of wildlife videos. In: *International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)* Bd. 6, 2005
- [**Cangar u. a. 2008**] CANGAR, Ö. ; LEROY, T. ; GUARINO, M. ; VRANKEN, E. ; FALLON, R. ; LENEHAN, J. ; MEE, J. ; BERCKMANS, D.: Automatic real-time monitoring of locomotion and posture behaviour of pregnant cows prior to calving using online image analysis. In: *Computers and Electronics in Agriculture (COMPAG)* 64 (2008), November, Nr. 1, 53 – 60. – DOI 10.1016/j.compag.2008.05.014. – ISSN 0168-1699
- [**Cha 2007**] CHA, Sung-Hyuk: Comprehensive survey on distance/similarity measures between probabilitydensity functions. In: *International Journal of Mathematical Models and Methods in Applied Sciences* 1 (2007), Nr. 4, 300 – 307
- [**Chai u. a. 2004**] CHAI, Xiujuan ; SHAN, Shiguang ; GAO, Wen ; LIU, Xin: Pose normalization using generic 3D face model as a priori for pose-insensitive face recognition. In: *Chinese Conference on Advances in Biometric Person Authentication (SINOBIOMETRICS)* Bd. 5. Berlin, Heidelberg : Springer, 2004, 144 – 152 – ISBN 978-3-5402-4029-7
- [**Chai u. a. 2005**] CHAI, Xiujuan ; SHAN, Shiguang ; GAO, Wen ; LIU, Xin: Pose normalization using generic 3D face model as a priori for pose-insensitive face recognition. In: LI, StanZ. (Hrsg.) ; LAI, Jianhuang (Hrsg.) ; TAN, Tieniu (Hrsg.) ; FENG, Guocan (Hrsg.) ; WANG, Yunhong (Hrsg.): *Advances in Biometric Person Authentication* Bd. 3338. Berlin, Heidelberg : Springer, 2005, 144 – 152. – DOI 10.1007/978-3-540-30548-4_17. – ISBN 978-3-5402-4029-7

- [**Chamoso u. a. 2014**] CHAMOSO, Pablo ; RAVEANE, William ; PARRA, Victor ; GONZÁLEZ, Angélica: UAVs applied to the counting and monitoring of animals. In: RAMOS, Carlos (Hrsg.) ; NOVAIS, Paulo (Hrsg.) ; NIHAN, Céline E. (Hrsg.) ; CORCHADO RODRÍGUEZ, Juan M. (Hrsg.): *Ambient Intelligence - Software and Applications (ISAmI)* Bd. 291. Berlin, Heidelberg : Springer, 2014, 71 – 80. – DOI 10.1007/978-3-319-07596-9_8. – ISBN 978-3-3190-7595-2
- [**Chang u. a. 2003**] CHANG, Kyong I. ; BOWYER, Kevin W. ; FLYNN, Patrick J.: Face recognition using 2D and 3D facial data. In: *ACM Workshop on Multimodal User Authentication*, 2003, 25 – 32.
- [**Chang u. a. 2005**] CHANG, Kyong I. ; BOWYER, Kevin W. ; FLYNN, Patrick J.: An evaluation of multimodal 2D+3D face biometrics. In: *Pattern Analysis and Machine Intelligence (TPAMI)* 27 (2005), April, Nr. 4, 619 – 624. – DOI 10.1109/TPAMI.2005.70. – ISSN 0162-8828
- [**Chanvichitkul u. a. 2007**] CHANVICHITKUL, M. ; KUMHOM, P. ; CHAMNONGTHAI, K.: Face recognition based dog breed classification using coarse-to-fine concept and PCA. In: *Asia-Pacific Conference on Communications (APCC)*, 2007, 25 – 29. – DOI 10.1109/APCC.2007.4433495
- [**Chen u. a. 2012a**] CHEN, Juan ; WEN, Quan ; QU, Wenming ; METE, Mutlu: Panda facial region detection based on topology modelling. In: *Image and Signal Processing (CISP)* Bd. 5, 2012, 911 – 915. – DOI 10.1109/CISP.2012.6469668
- [**Chen u. a. 2012b**] CHEN, Juan ; WEN, Quan ; ZHUO, Chenglong ; METE, Mutlu: A novel approach towards head detection of giant pandas in the free-range environment. In: *Image and Signal Processing (CISP)* Bd. 5, 2012, 814 – 818. – DOI 10.1109/CISP.2012.6469751
- [**Chen u. a. 2010**] CHEN, L. C. ; NGUYEN, X. L. ; LIANG, C. W.: Object segmentation method using depth slicing and region growing algorithms. In: *3D Systems and Applications (3DSA)* Bd. 2, 2010, 19 – 21. – Tokyo, Japan
- [**Chen u. a. 2003**] CHEN, Yu-Jen ; JEN, Sun-Lun ; LI, Yan-Chay ; YOUNG, M.S.: The implementation of a high resolution digital color image processing system for animal behavior measurement. In: *Asian-Pacific Conference on Biomedical Engineering (APBME)*, 2003, 298 – 299. – DOI 10.1109/APBME.2003.1302702
- [**Chen u. a. 2005**] CHEN, Yu-Jen ; LI, Yan-Chay ; HUANG, Ke-Nung ; YOUNG, Ming-Shing: The implementation of a stand-alone video tracking and analysis system for animal behavior measurement in morris water maze. In: *Engineering in Medicine and Biology Society (EMBC)*, 2005, 1766 – 1768. – DOI 10.1109/IEMBS.2005.1616788
- [**Chernov u. Lesort 2005**] CHERNOV, N. ; LESORT, C.: Least squares fitting of circles. In: *Mathematical Imaging and Vision (JMIV)* 23 (2005), Nr. 3, 239 – 252. – DOI 10.1007/s10851-005-0482-8. – ISSN 0924-9907

- [**Chow u. a. 2012**] CHOW, J. C. . ; ANG, K. D. ; LICHTI, D. D. ; TESKEY, W. F.: Performance analysis of a low-cost triangulation-based 3d camera: Microsoft Kinect system. In: *Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS) XXXIX-B5* (2012), 175 – 180. – DOI 10.5194/isprsarchives-XXXIX-B5-175-2012
- [**Chumerin u. van Hulle 2008**] CHUMERIN, Nikolay ; HULLE, Marc M.: Ground plane estimation based on dense stereo disparity. In: *Neural Networks and Artificial Intelligence (ICNNAI)* Bd. 5, 2008, 209 – 213. – Minsk, Belarus
- [**Chung u. a. 2014**] CHUNG, Yongwha ; KIM, Haelyeon ; LEE, Hansung ; PARK, Daihee ; JEON, Taewoong ; CHANG, Hong-Hee: A cost-effective pigsty monitoring system based on a video sensor. In: *Transactions on Internet and Information Systems (TIIS)* 8 (2014), Nr. 4, 1481 – 1498. – DOI 10.3837/tiis.2014.04.018
- [**Cohen 1996**] COHEN, Harvey A.: Parallel algorithm for gray-scale image segmentation. In: NARASIMHAN, V. L. (Hrsg.) ; JAIN, Lakhmi C. (Hrsg.): *Australian and New Zealand Conference on Intelligent Information Systems (ANZIIS)*, IEEE, 1996, 143 – 146 – DOI 10.1109/ANZIIS.1996.573919. – ISBN 0-7803-3667-4
- [**Combes u. a. 2008**] COMBES, B. ; HENNESSY, R. ; WADDINGTON, J. ; ROBERTS, N. ; PRIMA, S.: Automatic symmetry plane estimation of bilateral objects in point clouds. In: *Computer Vision and Pattern Recognition (CVPR)*, 2008. – DOI 10.1109/CVPR.2008.4587605. ISSN 1063-6919
- [**Comon 1994**] COMON, Pierre: Independent component analysis, a new concept? In: *Signal Processing* 36 (1994), April, Nr. 3, 287 – 314. – DOI 10.1016/0165-1684(94)90029-9. – ISSN 0165-1684
- [**Corkery u. a. 2007**] CORKERY, Gerard ; GONZALES-BARRON, Ursula A. ; BUTLER, Francis ; McDONNELL, Kevin ; WARD, Shane ; AGRICULTURAL, The American S. (Hrsg.) ; ENGINEERS, Biological (Hrsg.): *A preliminary investigation on face recognition as a biometric identifier of sheep*. American Society of Agricultural and Biological Engineers, 2007, 313 – 320.
- [**Coulon u. a. 2009**] COULON, Marjorie ; DEPUTTE, Bertrand L. ; HEYMAN, Yvan ; BAUDOIN, Claude: Individual recognition in domestic cattle (bos taurus): Evidence from 2D-images of heads from different breeds. In: *PLoS ONE, Public Library of Science* 4 (2009), Februar, Nr. 2. – DOI 10.1371/journal.pone.0004441
- [**Cover u. Hart 1967**] COVER, T. ; HART, P.: Nearest neighbor pattern classification. In: *Transactions on Information Theory (TIT)* 13 (1967), Januar, Nr. 1, 21 – 27. – DOI 10.1109/TIT.1967.1053964. – ISSN 0018-9448
- [**Dabarera u. Rodrigo 2010**] DABARERA, R. ; RODRIGO, R.: Vision based elephant recognition for management and conservation. In: *Information and Automation for Sustainability (ICIAfS)* Bd. 5, 2010, 163 – 166. – DOI 10.1109/ICIAFS.2010.5715653

- [**Dalal u. Triggs 2005**] DALAL, N. ; TRIGGS, B.: Histograms of oriented gradients for human detection. In: *Computer Vision and Pattern Recognition (CVPR)* Bd. 1, 2005, 886 – 893. – DOI 10.1109/CVPR.2005.177. – ISSN 1063-6919
- [**Davis u. Goadrich 2006**] DAVIS, Jesse ; GOADRIC, Mark: The relationship between precision-recall and ROC curves. In: *Machine Learning (ICML)* Bd. 23. New York, NY, USA : ACM, 2006, 233 – 240 – DOI 10.1145/1143844.1143874. – ISBN 1-5959-3383-2
- [**Dawkins u. a. 2012**] DAWKINS, Marian S. ; CAIN, Russell ; ROBERTS, Stephen J.: Optical flow, flock behaviour and chicken welfare. In: *Animal Behaviour* 84 (2012), Nr. 1, 219 – 223. – DOI 10.1016/j.anbehav.2012.04.036. – ISSN 0003-3472
- [**Dawkins u. a. 2009**] DAWKINS, Marian S. ; LEE, Hyoung-Joo ; WAITT, Corri D. ; ROBERTS, Stephen J.: Optical flow patterns in broiler chicken flocks as automated measures of behaviour and gait. In: *Applied Animal Behaviour Science* 119 (2009), Nr. 3, 203 – 209. – DOI 10.1016/j.applanim.2009.04.009
- [**DeCann u. Ross 2013**] DECANN, Brian ; ROSS, Arun: Relating ROC and CMC curves via the biometric menagerie. In: *Biometrics: Theory, Applications and Systems (BTAS)* Bd. 6 IEEE, 2013
- [**Delac u. Grgic 2004**] DELAC, K. ; GRGIC, M.: A survey of biometric recognition methods. In: *Electronics in Marine (Elmar)* Bd. 46, 2004, 184 – 193. – ISSN 1334-2630
- [**Delac u. a. 2005**] DELAC, Kresimir ; GRGIC, Mislav ; GRGIC, Sonja: Independent comparative study of PCA, ICA, and LDA on the FERET data set. In: *International Journal of Imaging Systems and Technology* 15 (2005), Nr. 5, 252 – 260. – DOI 10.1002/ima.20059
- [**DeMello 2012**] DEMELLO, M.: *Faces around the world: A cultural encyclopedia of the human face*. ABC-CLIO, 2012. – ISBN 978-1-5988-4617-1
- [**von den Driesch 1976**] DRIESCH, Angela von d. ; 2 (Hrsg.): *Das Vermessen von Tierknochen aus vor-und frühgeschichtlichen Siedlungen*. Institut für Paläoanatomie Domestikationsforschung und Geschichte der Tiermedizin, 1976
- [**Ernst u. Kublbeck 2011**] ERNST, A ; KUBLBECK, C.: Fast face detection and species classification of African great apes. In: *Advanced Video and Signal-Based Surveillance (AVSS)* Bd. 8, 2011, 279 – 284. – DOI 10.1109/AVSS.2011.6027337
- [**Farin 1996**] FARIN, Gerald E.: *Curves and surfaces for computer-aided geometric design: A practical code*. 4th. Orlando, FL, USA : Academic Press, Inc., 1996. – ISBN 0-1224-9054-1

- [Faugeras u. a. 2004] FAUGERAS, O. ; LUONG, Q.T. ; PAPADOPOULOU, T.: *The geometry of multiple images: The laws that govern the formation of multiple images of a scene and some of their applications*. MIT Press, 2004. – ISBN 978-0-2625-6204-1
- [Fawcett 2004] FAWCETT, Tom: ROC graphs: Notes and practical considerations for researchers / HP Laboratories. Mill Road, Palo Alto, CA 94304, März 2004 (ROC101). – Technical Report
- [Fawcett 2006] FAWCETT, Tom: An introduction to {ROC} analysis. In: *Pattern Recognition Letters (PRL)* 27 (2006), Nr. 8, 861 – 874. – DOI 10.1016/j.patrec.2005.10.010. – ISSN 0167-8655
- [Ferrari u. a. 2008] FERRARI, Vittorio ; FEVRIER, Loic ; JURIE, Frederic ; SCHMID, Cordelia: Groups of adjacent contour segments for object detection. In: *Pattern Analysis and Machine Intelligence (TPAMI)* 30 (2008), Nr. 1, 36 – 51
- [Ferrari u. a. 2010] FERRARI, Vittorio ; JURIE, Frederic ; SCHMID, Cordelia: From images to shape models for object detection. In: *Computer Vision (ICCV)* 87 (2010), Nr. 3, 284 – 303
- [Fisher 1936] FISHER, R. A.: The use of multiple measurements in taxonomic problems. In: *Annals of Eugenics* 7 (1936), Nr. 2, 179 – 188. – DOI 10.1111/j.1469-1809.1936.tb02137.x. – ISSN 2050-1439
- [Fleuret u. Geman 2008] FLEURET, F. ; GEMAN, D.: Stationary features and cat detection. In: *Journal of Machine Learning Research (JMLR)* 9 (2008), 2549 – 2578
- [Gafurov 2007] GAFUROV, Davrondzhon: A survey of biometric gait recognition: Approaches, security and challenges. In: *Annual Norwegian Computer Science Conference*, 2007, 19 – 21.
- [Gamble u. a. 2008] GAMBLE, Lloyd ; RAVELA, Sai ; MCGARIGAL, Kevin: Multi-scale features for identifying individuals in large biological databases: An application of pattern recognition technology to the marbled salamander *ambystoma opacum*. In: *Applied Ecology* 45 (2008), Nr. 1, 170 – 180
- [van Gemert u. a. 2014] GEMERT, J.C. van ; VERSCHOOR, C. ; METTES, P. ; EPEMA, K. ; KOH, L.P. ; WICH, S.: Nature conservation drones for automatic localization and counting of animals. In: *European Conference on Computer Vision (ECCV) Workshop*, 2014
- [Gibson u. a. 2003] GIBSON, David P. ; CAMPBELL, Neill W. ; THOMAS, Barry T.: Quadruped gait analysis using sparse motion information. In: *Image Processing (ICIP)* Bd. 3, 2003, 333 – 336 – DOI 10.1109/ICIP.2003.1247249. – ISBN 0-7803-7750-8

- [**Gilman u. a. 2013**] GILMAN, A ; DONG, Ting ; HUPMAN, K. ; STOCKIN, K. ; PAWLEY, M.: Dolphin fin pose correction using ICP in application to photo-identification. In: *Image and Vision Computing New Zealand (IVCNZ)* Bd. 28, 2013, 388 – 393. – DOI 10.1109/IVCNZ.2013.6727046. – ISSN 2151-2191
- [**Gleerup u. a. 2015**] GLEERUP, Karina B. ; FORKMAN, Björn ; LINDEGAARD, Casper ; ANDERSEN, Pia H.: An equine pain face. In: *Veterinary Anaesthesia and Analgesia* 42 (2015), Nr. 1, 103 – 114. – DOI 10.1111/vaa.12212. – ISSN 1467-2995
- [**Goel u. Nehra 2011**] GOEL, Tripti ; NEHRA, Vijay: Comparative analysis of various illumination normalization techniques for face recognition. In: *Computer Applications (IJCA)* 28 (2011), August, Nr. 9
- [**Gökberk u. a. 2009**] GÖKBERK, Berk ; SALAH, Albert A. ; AKARUN, Lale ; ETHEVE, Remy ; RICCIO, Daniel ; DUGELAY, Jean-Luc: 3D face recognition. In: *Guide to Biometric Reference Systems and Performance Evaluation*. Berlin, Heidelberg : Springer, 2009, 263 – 295.
- [**Gonzalez u. Woods 2008**] GONZALEZ, Rafael C. ; WOODS, Richard E.: *Digital image processing*. 3rd. Upper Saddle River, NJ, USA : Prentice-Hall, Inc., 2008, 655 – 657. – ISBN 978-0-1316-8728-8
- [**Goudelis u. a. 2008**] GOUDELIS, Georgios ; TEFAS, Anastasios ; PITAS, Ioannis: Emerging biometric modalities: A survey. In: *Multimodal User Interfaces 2* (2008), Nr. 3-4, 217 – 235
- [**Gupta u. a. 2007**] GUPTA, Shalini ; MARKEY, Mia K. ; BOVIK, Alan C.: Advances and challenges in 3D and 2D+3D human face recognition. In: *Pattern Recognition in Biology* (2007), 63 – 103
- [**Han u. a. 2013**] HAN, J. ; SHAO, L. ; XU, D. ; SHOTTON, J.: Enhanced computer vision with Microsoft Kinect sensor: A review. In: *Cybernetics* 43 (2013), Oktober, Nr. 5, 1318 – 1334. – DOI 10.1109/TCYB.2013.2265378
- [**Hannuna 2007**] HANNUNA, Sion: *Quadruped gait detection in low quality wildlife video*, University of Bristol, PhD thesis, Dezember 2007
- [**Hannuna u. a. 2005**] HANNUNA, S.L. ; CAMPBELL, N.W. ; GIBSON, D.P.: Segmenting quadruped gait patterns from wildlife video. In: *Visual Information Engineering (VIE)* (2005), Januar, 235 – 242. – DOI 10.1049/cp_20050095
- [**Hartley u. Zisserman 2004**] HARTLEY, R. I. ; ZISSERMAN, A.: *Multiple view geometry in computer vision*. 2nd. Cambridge University Press, 2004. – ISBN 0-5215-4051-8
- [**Herrera u. a. 2011**] HERRERA, Daniel C. ; KANNALA, Juho ; HEIKKIL, Janne: Accurate and practical calibration of a depth and color camera pair. In: *Computer Analysis of Images and Patterns (CAIP)* 6855 (2011), 437 – 445

- [**Herrera u. a. 2012**] HERRERA, Daniel C. ; KANNALA, Juho ; HEIKKIL, Janne: Joint depth and color camera calibration with distortion correction. In: *Pattern Analysis and Machine Intelligence (TPAMI)* 34 (2012), Oktober, Nr. 10, 2058 – 2064. – DOI 10.1109/TPAMI.2012.125. – ISSN 0162–8828
- [**van Hertem u. a. 2013**] HERTEM, T. van ; ALCHANATIS, V. ; ANTLER, A. ; MALTZ, E. ; HALACHMI, I. ; SCHLAGETER-TELLO, A. ; LOKHORST, C. ; VIAZZI, S. ; ROMANINI, C.E.B. ; PLUK, A. ; BAHR, C. ; BERCKMANS, D.: Comparison of segmentation algorithms for cow contour extraction from natural barn background in side view images. In: *Computers and Electronics in Agriculture (COMPAG)* 91 (2013), 65 – 74. – DOI 10.1016/j.compag.2012.12.003. – ISSN 0168–1699
- [**van Hertem u. a. 2014**] HERTEM, T. van ; VIAZZI, S. ; STEENSELS, M. ; MALTZ, E. ; ANTLER, A. ; ALCHANATIS, V. ; SCHLAGETER-TELLO, A. A. ; LOKHORST, K. ; ROMANINI, E. C. ; BAHR, C. u. a.: Automatic lameness detection based on consecutive 3D-video recordings. In: *Biosystems Engineering* 119 (2014), 108 – 116
- [**Hetzel u. a. 2001**] HETZEL, Guenter ; LEIBE, Bastian ; LEVI, Paul ; SCHIELE, Bernt: 3D object recognition from range images using local feature histograms. In: *Computer Vision and Pattern Recognition (CVPR)*, 2001, 394 – 399.
- [**Hiby u. a. 2009**] HIBY, Lex ; LOVELL, Phil ; PATIL, Narendra ; KUMAR, N S. ; GOPALASWAMY, Arjun M. ; KARANTH, K U.: A tiger cannot change its stripes: using a three-dimensional model to match images of living tigers and tiger skins. In: *Biology Letters* 5 (2009), Juni, Nr. 3, 383 – 386. – DOI 10.1098/rsbl.2009.0028
- [**Hillman u. a. 2003**] HILLMAN, GR ; WURSIG, B ; GAILEY, GA ; KEHTARNAVAZ, N ; DROBYSHEVSKY, A ; ARAABI, BN ; TAGARE, HD ; WELLER, DW: Computer-assisted photo-identification of individual marine vertebrates: A multi-species system. In: *Aquatic Mammals* 29 (2003), Nr. 1, 117 – 123
- [**Hinrichs Innovation + Technik GmbH 2012**] HINRICHS INNOVATION + TECHNIK GMBH, HIT: *Interne Dokumentation HIT*. 2012-2013. – Kleine Straße 2, D-25795 Weddingstedt
- [**Hirsch u. a. 1952**] HIRSCH, Morris ; GRAHAM, Edmund F. ; DRACY, Arthur E.: A classification for the identification of bovine noseprints. In: *Dairy Science (JDS)* 35 (1952), Nr. 4, 314 – 319. – DOI 10.3168/jds.S0022–0302(52)93707–7. – ISSN 0022–0302
- [**Hjelmås u. Low 2001**] HJELMÅS, Erik ; LOW, Boon K.: Face detection: A survey. In: *Computer Vision and Image Understanding (CVIU)* 83 (2001), Nr. 3, 236 – 274. – DOI 10.1006/cviu.2001.0921. – ISSN 1077–3142

- [**Hoque u. a. 2011**] HOQUE, Sanaul ; AZHAR, M.A. Hannan B. ; DERAVID, Farzin:
ZOOMETRICS - Biometric identification of wildlife using natural body marks.
In: *Bio-Science and Bio-Technology* Bd. 3, 2011, 45 – 53. – ISSN 2233–7849
- [**Horn 1987**] HORN, Berthold K. P.: Closed-form solution of absolute orientation
using unit quaternions. In: *Optical Society of America A (JOSA A)* 4 (1987),
Nr. 4, 629 – 642. – DOI 10.1364/JOSAA.4.000629
- [**Hough 1962**] HOUGH, Paul: *Method and means for recognizing complex patterns*.
U.S. Patent 3.069.654. Version: Dezember 1962
- [**Howland u. a. 2004**] HOWLAND, Howard C. ; MEROLA, Stacey ; BASARAB, Jen-
nifer R.: The allometry and scaling of the size of vertebrate eyes. In: *Vision
Research* 44 (2004), Nr. 17, 2043 – 2065. – DOI 10.1016/j.visres.2004.03.023. –
ISSN 0042–6989
- [**Hu u. Xin 2000**] HU, J. ; XIN, H.: Image-processing algorithms for behavior
analysis of group-housed pigs. In: *Behavior Research Methods, Instruments,
Computers* 32 (2000), Februar, Nr. 1, 72 – 85
- [**Huang u. a. 2011**] HUANG, Di ; SHAN, Caifeng ; ARDABILIAN, M. ; WANG,
Yunhong ; CHEN, Liming: Local binary patterns and its application to facial
image analysis: A survey. In: *Systems, Man, and Cybernetics (SMC)* 41 (2011),
November, Nr. 6, 765 – 781. – DOI 10.1109/TSMCC.2011.2118750. – ISSN
1094–6977
- [**Huang u. a. 2007**] HUANG, Gary B. ; JAIN, Vidit ; LEARNED-MILLER, Erik:
Unsupervised joint alignment of complex images. In: *Computer Vision (ICCV)*
Bd. 11 IEEE, 2007
- [**Hughes u. a. 2013**] HUGHES, Benjamin J ; SCHOLL, Michael ; CUTHILL, Innes C
; BURGHARDT, Tilo: Fin-printing great white sharks *Carcharodon carcharias*:
A fully automated computer vision system for the identification of individual
animals. In: *Conference on Behaviour, Physiology and Genetics of Wildlife* 9
(2013), 93 – 106
- [**Huynh 2009**] HUYNH, DuQ.: Metrics for 3D rotations: Comparison and analysis.
In: *Mathematical Imaging and Vision (JMIV)* 35 (2009), Nr. 2, 155 – 164. –
DOI 10.1007/s10851-009-0161-2. – ISSN 0924–9907
- [**IDS 2013**] IDS: *Camera and sensor data UI-225x/UI-625x*. Kameraspezifikation
- IDS Imaging - Website. Version: Februar 2013. – Version 4.20 (20. Feb. 2013)
- [**Illingworth u. Kittler 1988**] ILLINGWORTH, J. ; KITTLER, J.: A survey of
the hough transform. In: *Computer Vision, Graphics, and Image Processing*
44 (1988), Nr. 1, 87 – 116. – DOI 10.1016/S0734-189X(88)80033-1. – ISSN
0734–189X

- [**Irsoy u. a. 2012**] IRSOY, Ozan ; YILDIZ, Olcay T. ; ALPAYDIN, Ethem: Design and analysis of classifier learning experiments in bioinformatics: Survey and case studies. In: *Computational Biology and Bioinformatics (BCB)* 9 (2012), Nr. 6, 1663 – 1675
- [**Ishii u. a. 2004**] ISHII, Yohei ; HONGO, Hitoshi ; YAMAMOTO, Kazuhiko ; NIWA, Yoshinori: Real-time face and head detection using four directional features. In: *Automatic Face and Gesture Recognition (FG)* Bd. 6 IEEE, 2004, 403 – 408.
- [**Ismayilova 2014**] ISMAYILOVA, Gunel: *The use of image labelling to identify pig behaviours for the development of a real-time monitoring and control tool*, Università degli Studi di Milano, PhD thesis, Mai 2014
- [**Jaccard 1901**] JACCARD, Paul: Étude comparative de la distribution florale dans une portion des alpes et des jura. In: *Bulletin del la Société Vaudoise des Sciences Naturelles* 37 (1901), 547 – 579
- [**Jafri u. Arabnia 2009**] JAFRI, Rabia ; ARABNIA, Hamid R.: A Survey of face recognition techniques. In: *Information Processing Systems (JIPS)* 5 (2009), Nr. 2, 41 – 68. – DOI 10.3745/JIPS.2009.5.2.041
- [**Jain u. a. 2004**] JAIN, Anil K. ; ROSS, Arun ; PRABHAKAR, Salil: An introduction to biometric recognition. In: *Circuits and Systems for Video Technology (CSVT)* 14 (2004), Nr. 1, 4 – 20
- [**Jover u. a. 2009**] JOVER, J.M. N. ; RAYA, M. A. ; GÓMEZ, V. ; BALASCH, S. ; MORENO, J.R. ; COLOMER, V. G. ; TORRES, A.: An automatic colour-based computer vision algorithm for tracking the position of piglets. In: *Spanish Journal of Agricultural Research (SJAR)* 7 (2009), Nr. 3, 535 – 549. – ISSN 2171 – 9292
- [**Juan u. Gwun 2009**] JUAN, Luo ; GWUN, Oubong: A comparison of SIFT, PCA-SIFT and SURF. In: *Image Processing (IJIP)* 3 (2009), Nr. 4, 143 – 152
- [**Kahraman u. a. 2007**] KAHRAMAN, F. ; KURT, B. ; GOKMEN, M.: Robust face alignment for illumination and pose invariant face recognition. In: *Computer Vision and Pattern Recognition (CVPR)*, 2007. – DOI 10.1109/CVPR.2007.383400. ISSN 1063–6919
- [**Kaiyan u. a. 2012**] KAIYAN, Lin ; XUEJUN, Yang ; JUNHUI, Wu ; JIE, Chen ; HUIPING, Si: A real-time computer vision monitoring way for animal diversity. In: *World Automation Congress (WAC)*, 2012. – ISSN 2154–4824
- [**Kammer 2012**] KAMMER, M.: *Schwingungsspektroskopische Untersuchungen zur Biomineralisation*, Technische Universität Dresden, PhD thesis, Oktober 2012. – Fakultät Mathematik und Naturwissenschaften

- [**Kasa 1976**] KASA, I.: A circle fitting procedure and its error analysis. In: *Instrumentation and Measurement (TIM)* IM-25 (1976), März, Nr. 1, 8 – 14. – DOI 10.1109/TIM.1976.6312298. – ISSN 0018–9456
- [**Kashiha u. a. 2013**] KASHIHA, Mohammadamin ; BAHR, Claudia ; OTT, Sanne ; MOONS, Christel P. ; NIEWOLD, Theo A. ; ÖDBERG, F.O. ; BERCKMANS, Daniel: Automatic identification of marked pigs in a pen using image pattern recognition. In: *Computers and Electronics in Agriculture (COMPAG)* 93 (2013), 111 – 120. – DOI 10.1016/j.compag.2013.01.013. – ISSN 0168–1699
- [**Kelly 2001**] KELLY, Marcella J.: Computer-aided photograph matching in studies using individual identification: An example from serengeti cheetahs. In: *Journal of Mammalogy* 82 (2001), Nr. 2, 440 – 449
- [**Kendrick u. Feng 2011**] KENDRICK, Keith M. ; FENG, Jianfeng: Neural encoding principles in face perception revealed using non-primatemodels. In: CALDER, Andy (Hrsg.) ; JOHNSON, Mark (Hrsg.) ; HAXBY, James V. (Hrsg.): *Oxford Handbook of Face Perception*. Oxford University Press, Juli 2011. – DOI 10.1093/oxfordhob/9780199559053.013.0034. – ISBN 978–0–1995–5905–3
- [**Khan u. a. 2013**] KHAN, Ijaz ; ABDULLAH, Hadi ; ZAINAL, Mohd Shamian B.: Efficient eyes and mouth detection algorithm using combination of viola jones and skin color pixel detection. In: *International Journal of Engineering* 3 (2013), Nr. 4, 51 – 60
- [**Khan u. a. 2009**] KHAN, Maleika Heenaye-Mamode ; KHAN, Naushad M. ; SUBRAMANIAN, Raja K.: Feature extraction of dorsal hand vein pattern using a fast modified PCA algorithm based on cholesky decomposition and lanczos technique. In: *Mathematical and Computer Sciences (IJMCS)* 5 (2009), Nr. 4, 230 – 234
- [**Kholmatov u. Yanikoglu 2005**] KHOLMATOV, Alisher ; YANIKOGLU, Berrin: Identity authentication using improved online signature verification method. In: *Pattern Recognition Letters (PRL)* 26 (2005), Nr. 15, 2400 – 2408
- [**Khoramshahi u. a. 2014**] KHORAMSHAHI, Ehsan ; HIETAOJA, Juha ; VALROS, Anna ; YUN, Jinhyeon ; PASTELL, Matti: Real-time recognition of sows in video: A supervised approach. In: *Information Processing in Agriculture (IPA)* 1 (2014), Nr. 1, 73 – 81. – DOI 10.1016/j.inpa.2014.07.002. – ISSN 2214–3173
- [**Khorrami u. a. 2012**] KHORRAMI, P ; WANG, J ; HUANG, T: Multiple animal species detection using robust principal component analysis and large displacement optical flow. In: *Pattern Recognition (ICPR) Workshop* 21 (2012), November
- [**Khoshelham 2011**] KHOSHELHAM, Kourosh: Accuracy analysis of Kinect depth data. In: *Proceedings (ISPRS) Workshop Laser scanning* Bd. 38, 2011

- [**Khoshelham u. Elberink 2012**] KHOSHELHAM, Kourosh ; ELBERINK, Sander O.: Accuracy and resolution of Kinect depth data for indoor mapping applications. In: *Sensors* 12 (2012), Nr. 2, 1437 – 1454. – DOI 10.3390/s120201437
- [**Kim u. a. 2005a**] KIM, Hyeon T. ; CHOI, Hong L. ; LEE, Dae W. ; YOON, Yong C.: Recognition of individual holstein cattle by imaging body patterns. In: *Asian-Australasian Journal of Animal Sciences (AJAS)* 18 (2005), Nr. 8, 1194 – 1198
- [**Kim u. a. 2005b**] KIM, Hyeon T. ; IKEDA, Y. ; CHOI, Hong L.: The identification of japanese black cattle by their faces. In: *Asian-Australasian Journal of Animal Sciences (AJAS)* 18 (2005), Nr. 6, 868 – 872
- [**Kim u. Jung 2013**] KIM, Jieun ; JUNG, Woo Y.: Vision-based cattle detection and localization system in an RGB color space. In: *Intelligent Systems and Applications (INTELLI)* Bd. 2, 2013, 173 – 175.
- [**Kirby u. Sirovich 1990**] KIRBY, M. ; SIROVICH, L.: Application of the karhunen-loeve procedure for the characterization of human faces. In: *Pattern Analysis and Machine Intelligence (TPAMI)* 12 (1990), Januar, Nr. 1, 103 – 108. – DOI 10.1109/34.41390. – ISSN 0162-8828
- [**Kohavi 1995**] KOHAVI, Ron: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: *International Joint Conference on Artificial Intelligence (IJCAI)* Bd. 2. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc., 1995, 1137 – 1143 – ISBN 1-5586-0363-8
- [**Koik u. Ibrahim 2012**] KOIK, Boon T. ; IBRAHIM, Haidi: A literature survey on animal detection methods in digital images. In: *Future Computer and Communication (IJFCC)* 1 (2012), Juni, Nr. 1
- [**Kolda u. a. 2003**] KOLDA, Tamara G. ; LEWIS, Robert M. ; TORCZON, Virginia: Optimization by direct search: New perspectives on some classical and modern methods. In: *SIAM Review (Society for Industrial and Applied Mathematics)* 45 (2003), 385 – 482
- [**Kong u. a. 2009**] KONG, Adams ; ZHANG, David ; KAMEL, Mohamed: A survey of palmprint recognition. In: *Pattern Recognition* 42 (2009), Nr. 7, 1408 – 1418
- [**Kong u. a. 2005**] KONG, Seong G. ; HEO, Jingu ; ABIDI, Bisma R. ; PAIK, Joonki ; ABIDI, Mongi A.: Recent advances in visual and infrared face recognition—a review. In: *Computer Vision and Image Understanding (CVIU)* 97 (2005), Nr. 1, 103 – 135
- [**Kongsro 2014**] KONGSRO, Jorgen: Estimation of pig weight using a Microsoft Kinect prototype imaging system. In: *Computers and Electronics in Agriculture (COMPAG)* 109 (2014), Nr. 0, 32 – 35. – DOI 10.1016/j.compag.2014.08.008. – ISSN 0168-1699

- [**Kouda u. a. 2011**] KOUDA, Mayumi ; MORIMOTO, Masakazu ; FUJII, Kensaku: A face identification method of non-native animals for intelligent trap. In: *Machine Vision Applications (MVA)*, 2011, 426 – 429.
- [**Kozakaya u. a. 2009**] KOZAKAYA, T. ; ITO, S. ; KUBOTA, S. ; YAMAGUCHI, O.: Cat face detection with two heterogeneous features. In: *Image Processing (ICIP)*, 2009, 1213 – 1216. – DOI 10.1109/ICIP.2009.5413669. – ISSN 1522-4880
- [**Krummenauer u. a. 2007**] KRUMMENAUER, F. ; WOJCIECHOWSKI, C. ; BAULIG, C. ; AL-NAWAS, B.: Boxplots – die flexible Alternative zum „Antennen-Bildchen“. In: *Zeitschrift für Zahnärztliche Implantologie (ZZI)* 23 (2007), 308 – 310
- [**Krutsch u. Tenorio 2011**] KRUTSCH, Robert ; TENORIO, David: Histogram equalization / Microcontroller Solutions Group Guadalajara. 2011 (Document Number: AN4318). – Freescale Semiconductor Application Note
- [**Kühl u. Burghardt 2013**] KÜHL, Hjalmar S. ; BURGHARDT, Tilo: Animal biometrics: Quantifying and detecting phenotypic appearance. In: *Trends in Ecology & Evolution* 28 (2013), Nr. 7, 432 – 441. – DOI 10.1016/j.tree.2013.02.013. – ISSN 0169-5347
- [**Kumar u. a. 2003**] KUMAR, Ajay ; WONG, David C. ; SHEN, Helen C. ; JAIN, Anil K.: Personal verification using palmprint and hand geometry biometric. In: *Audio-and Video-Based Biometric Person Authentication (AVBPA)*. Berlin, Heidelberg, 2003, 668 – 678.
- [**Kumar u. Singh 2014**] KUMAR, S. ; SINGH, S: Biometric recognition for pet animal. In: *Software Engineering and Applications (JSEA)* 7 (2014), Mai, Nr. 5, 470 – 482. – DOI 10.4236/jsea.2014.75044
- [**Kumar u. Divya 2014**] KUMAR, Y.H. S. ; DIVYA, C.D.: Feature selection approach in animal classification. In: *Signal & Image Processing (SIPIJ)* 5 (2014), August, Nr. 4, 55 – 66. – DOI 10.5121/sipij.2014.5406
- [**Lahiri u. a. 2011**] LAHIRI, Mayank ; TANTIPATHANANANDH, Chayant ; WARUNGU, Rosemary ; RUBENSTEIN, Daniel I. ; BERGER-WOLF, Tanya Y.: Biometric animal databases from field photographs: Identification of individual zebra in the wild. In: *Multimedia Retrieval (ICMR)* Bd. 1. New York, NY, USA : ACM, 2011. – DOI 10.1145/1991996.1992002. ISBN 978-1-4503-0336-1
- [**Larsson u. a. 2000**] LARSSON, Maria ; FINKEL, Deborah ; PEDERSEN, Nancy L.: Odor identification influences of age, gender, cognition, and personality. In: *The Journals of Gerontology Series B: Psychological Sciences and Social Sciences* 55 (2000), Nr. 5, 304 – 310

- [Lee u. Yoo 2014] LEE, Gyu-cheol ; YOO, Jisang: Real-time virtual-viewpoint image synthesis algorithm using Kinect camera. In: *Electrical Engineering & Technology (JEET)* Bd. 9, 2014, 1016 – 1022. – ISSN 1975–0102
- [Lee u. a. 2011] LEE, Hyoung-joo ; ROBERTS, Stephen J. ; DRAKE, Kelly A. ; DAWKINS, Marian S.: Prediction of feather damage in laying hens using optical flows and markov models. In: *Journal of The Royal Society Interface* 8 (2011), Nr. 57, 489 – 499. – DOI 10.1098/rsif.2010.0268. – ISSN 1742–5689
- [Lee u. Ho 2009] LEE, Sang-Beom ; HO, Yo-Sung: Discontinuity-adaptive depth map filtering for 3D view generation. In: *Immersive Telecommunications (IMMERSCOM)* Bd. 2. ICST, Brussels, Belgium, Belgium : ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2009. – ISBN 978–9–6397–9939–4
- [Lee u. Ho 2011] LEE, Sang-Beom ; HO, Yo-Sung: Real-time stereo view generation using Kinect depth camera. In: *Signal and Information Processing Association (APSIPA)*. Xi'an, China, 2011
- [Leroy u. a. 2008] LEROY, Toon ; BAHR, Claudia ; VRANKEN, Erik ; XIANGYU, Song ; MARTENS, Willem ; VANGHEYTE, Juergen ; NUFFEL, Annelies van ; SONCK, Bart ; BERCKMANS, Daniel: Automatic detection of lameness in dairy cattle–image features related to lameness. In: *Conference on Lameness in Ruminants*, 2008, 289 – 293.
- [Leroy u. a. 2006] LEROY, Toon ; VRANKEN, Erik ; BRECHT, Andres van ; STRUELENS, E ; SONCK, B ; BERCKMANS, Daniel: A computer vision method for on-line behavioral quantification of individually caged poultry. In: *Transactions of the ASAE* 49 (2006), Nr. 3, 795 – 802. – DOI 10.13031/2013.20462
- [Lévesque u. Bergevin 2010] LÉVESQUE, Olivier ; BERGEVIN, Robert: Detection and identification of animals using stereo vision. In: *Pattern Recognition (ICPR) Workshop*, 2010
- [Lewis u. Torczon 1999] LEWIS, R. ; TORCZON, V.: Pattern search algorithms for bound constrained minimization. In: *SIAM Journal on Optimization* 9 (1999), Nr. 4, 1082 – 1099. – DOI 10.1137/S1052623496300507
- [Li u. a. 2013] LI, B.Y.L. ; MIAN, AS. ; LIU, Wanquan ; KRISHNA, A: Using Kinect for face recognition under varying poses, expressions, illumination and disguise. In: *Applications of Computer Vision (WACV)*, 2013, 186 – 192. – DOI 10.1109/WACV.2013.6475017. – ISSN 1550–5790
- [Li u. a. 2006] LI, Ganhua ; CAI, Xuanping ; LI, Xianshuai ; LIU, Yunhui: An efficient face normalization algorithm based on eyes detection. In: *Intelligent Robots and Systems (RSJ)*, 2006, 3843 – 3848. – DOI 10.1109/IROS.2006.281791

- [**Lind u. a. 2005**] LIND, N.M. ; VINTHER, M. ; HEMMINGSEN, R.P. ; HANSEN, A.K.: Validation of a digital video tracking system for recording pig locomotor behaviour. In: *Journal of Neuroscience Methods* 143 (2005), Nr. 2, 123 – 132. – DOI 10.1016/j.jneumeth.2004.09.019
- [**Loos 2013**] LOOS, A: Chimpanzee identification using global and local features. In: *Acoustics, Speech and Signal Processing (ICASSP)*, 2013, 2347 – 2351. – DOI 10.1109/ICASSP.2013.6638074. – ISSN 1520–6149
- [**Loos u. Pfitzer 2012**] LOOS, A ; PFITZER, M.: Towards automated visual identification of primates using face recognition. In: *Systems, Signals and Image Processing (IWSSIP)*, 2012, 425 – 428. – ISSN 2157–8672
- [**Loos 2012**] LOOS, Alexander: Identification of great apes using gabor features and localitypreserving projections. In: *Multimedia Analysis for Ecological Data (MAED)*. New York, NY, USA : ACM, 2012, 19 – 24 – DOI 10.1145/2390832.2390838. – ISBN 978–1–4503–1588–3
- [**Loos u. Ernst 2012**] LOOS, Alexander ; ERNST, Andreas: Detection and identification of chimpanzee faces in the wild. In: *Multimedia (ISM)*. Washington, DC, USA : IEEE Computer Society, 2012, 116 – 119 – DOI 10.1109/ISM.2012.30. – ISBN 978–0–7695–4875–3
- [**Loos u. Ernst 2013**] LOOS, Alexander ; ERNST, Andreas: An automated chimpanzee identification system using face detection and recognition. In: *Journal on Image and Video Processing (EURASIP)* 49 (2013), Nr. 1. – DOI 10.1186/1687–5281–2013–49. – ISSN 1687–5281
- [**Loos u. a. 2011a**] LOOS, Alexander ; PFITZER, Martin ; APORIUS, Laura: Facial recognition for primate photo identification. In: *International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, 2011 (12)
- [**Loos u. a. 2011b**] LOOS, Alexander ; PFITZER, Martin ; APORIUS, Laura: Identification of great apes using face recognition. In: *European Signal Processing Conference (EUSIPCO)* Bd. 19, 2011, 922 – 926.
- [**López u. Ruiz 2010**] LÓPEZ, L.S. ; RUIZ, F.T.: *Local binary patterns applied to face detection and recognition*. 2010
- [**Lu u. a. 2003**] LU, Yongzhong ; ZHOU, Jingli ; YU, Shengsheng: A survey of face detection, extraction and recognition. In: *Computing and Informatics (CAI)* 22 (2003), Nr. 2, 163 – 195
- [**Majnik u. Bosnic 2013**] MAJNIK, Matjaz ; BOSNIC, Zoran: ROC analysis of classifiers in machine learning: A survey. In: *Intelligent Data Analysis* 17 (2013), Nr. 3, 531 – 558. – DOI 10.3233/IDA–130592

- [**Malassiotis u. Strintzis 2004**] MALASSIOTIS, S. ; STRINTZIS, M.G.: Pose and illumination compensation for 3D face recognition. In: *Image Processing (ICIP)* Bd. 1, 2004, 91 – 94. – DOI 10.1109/ICIP.2004.1418697. – ISSN 1522–4880
- [**Maltoni u. a. 2009**] MALTONI, Davide ; MAIO, Dario ; JAIN, Anil K. ; PRABHAKAR, Salil: *Handbook of fingerprint recognition*. Berlin, Heidelberg : Springer, 2009
- [**Marchant 2002**] MARCHANT, Jeanette: Secure animal identification and source verification. In: *JM Communications, UK. Copyright Optibrand Ltd., LLC* (2002)
- [**Markowitz u. a. 2003**] MARKOWITZ, Tim M. ; HARLIN, AprilD. ; WÜRSIG, Bernd: Digital photography improves efficiency of individual dolphin identification. In: *Marine Mammal Science* 19 (2003), Nr. 1, 217 – 223. – DOI 10.1111/j.1748–7692.2003.tb01103.x. – ISSN 1748–7692
- [**Martinez-Ortiz u. a. 2013**] MARTINEZ-ORTIZ, Carlos A. ; EVERSON, Richard M. ; MOTTRAM, Toby: Video tracking of dairy cows for assessing mobility scores. In: *Precision Livestock Farming (ECPLF)* 6th (2013), September
- [**McCool u. a. 2007**] MCCOOL, Chris ; MAMIC, George ; FOOKES, Clinton ; SRIDHARAN, Sridha: Normalisation of 3D face data. In: WYSOCKI, B (Hrsg.) ; WYSOCKI, T (Hrsg.): *Signal Processing and Communication Systems (CSPCS)*. Australia, Queensland, Gold Coast : DSP for Communication Systems, 2007, 17 – 19.
- [**Mian u. a. 2006**] MIAN, A ; BENNAMOUN, M. ; OWENS, R.: Automatic 3D face detection, normalization and recognition. In: *3D Data Processing, Visualization, and Transmission (3DPVT)* Bd. 3, 2006, 735 – 742. – DOI 10.1109/3DPVT.2006.32
- [**Micron 2006**] MICRON: *1/2-inch megapixel CMOS digital image sensor*. Datenblatt - Micron - Website. Version: Mai 2006. – Rev. F 5/06 EN (20. Feb. 2013)
- [**Minagawa u. a. 2002**] MINAGAWA, H ; FUJIMURA, T ; ICHIYANAGI, M ; TANAKA, K: Identification of beef cattle by analyzing images of their muzzle patterns lifted on paper. In: *Asian Conference for Information Technology in Agriculture (AFITA)* 8 (2002), 596 – 600
- [**Morrow-Tesch u. a. 1998**] MORROW-TESCH, J ; DAILEY, J ; JIANG, H: A video data base system for studying animal behavior. In: *Animal Science (JAS)* 76 (1998), Nr. 10, 2605 – 2608
- [**Murphy 2012**] MURPHY, Kevin P.: *Machine learning: A probabilistic perspective*. MIT Press, 2012. – ISBN 978–0–2620–1802–9

- [**Murphy-Chutorian u. Trivedi 2009**] MURPHY-CHUTORIAN, E. ; TRIVEDI, M.M.: Head pose estimation in computer vision: A survey. In: *Pattern Analysis and Machine Intelligence (TPAMI)* 31 (2009), April, Nr. 4, 607 – 626. – DOI 10.1109/TPAMI.2008.106. – ISSN 0162–8828
- [**Muruganantham u. Jebarajan 2012**] MURUGANANTHAM, S ; JEBARAJAN, T: A comprehensive review of significant researches on face recognition based on various conditions. In: *Computer Theory and Engineering (IJCTE)* 4 (2012), Nr. 1, 7 – 15
- [**Musgrave u. L. 2002**] MUSGRAVE, C. ; L., Cambier J.: *System and method of animal identification and animal transaction authorization using iris patterns*. US Patent. Version: Juli 2002. – US6424727 B1
- [**Nakarmi u. a. 2014**] NAKARMI, Akash D. ; TANG, Lie ; XIN, Hongwei: Automated tracking and behavior quantification of laying hens using 3D computer vision and radio frequency identification technologies. In: *American Society of Agricultural and Biological Engineers (ASABE)* 57 (2014), Nr. 5, 1455 – 1472. – DOI 10.13031/trans.57.10505
- [**Nakarmi 2013**] NAKARMI, Akash D.: *Automated inter-plant spacing sensing of corn plant seed lings and quantification of laying hen behaviors using 3D computer vision*. Ames, Iowa, Iowa State University, PhD thesis, 2013. – Paper 13313
- [**Niese u. a. 2007**] NIESE, Robert ; AL-HAMADI, Ayoub ; MICHAELIS, Bernd: A novel method for 3D face detection and normalization. In: *Multimedia (JMM)* 2 (2007), Nr. 5
- [**Niknam u. a. 2010**] NIKNAM, Mehdi ; THULASIRAMAN, Parimala ; CAMORLINGA, Sergio: A parallel algorithm for connected component labelling of gray-scale images on homogeneous multicore architectures. In: *Journal of Physics: Conference Series (JPCS)* 256 (2010), Nr. 1
- [**Noviyanto u. Arymurthy 2012**] NOVIYANTO, Ary ; ARYMURTHY, Aniati M.: Automatic cattle identification based on muzzle photo using speed-up robust features approach. In: *Computer Science (ECCS)* Bd. 110, 2012, 110 – 114 – ISBN 978–1–6180–4140–1
- [**Noviyanto u. Arymurthy 2013**] NOVIYANTO, Ary ; ARYMURTHY, Aniati M.: Beef cattle identification based on muzzle pattern using a matching refinement technique in the {SIFT} method. In: *Computers and Electronics in Agriculture (COMPAG)* 99 (2013), Nr. 0, 77 – 84. – DOI 10.1016/j.compag.2013.09.002. – ISSN 0168–1699
- [**Ojala u. a. 1994**] OJALA, T. ; PIETIKAINEN, M. ; HARWOOD, D.: Performance evaluation of texture measures with classification based on kullback discrimi-

- nation of distributions. In: *International Conference A: Pattern Recognition (IAPR)* Bd. 1, 1994, 582 – 585. – DOI 10.1109/ICPR.1994.576366
- [**Ojala u. a. 1996**] OJALA, Timo ; PIETIKÄINEN, Matti ; HARWOOD, David: A comparative study of texture measures with classification based on featured distributions. In: *Pattern Recognition* 29 (1996), Nr. 1, 51 – 59. – DOI 10.1016/0031-3203(95)00067-4. – ISSN 0031-3203
- [**Okada u. a. 2001**] OKADA, K. ; KAGAMI, S. ; INABA, M. ; INOUE, H.: Plane segment finder: Algorithm, implementation and applications. In: *Robotics and Automation (ICRA)*, 2001, 2120 – 2125. – DOI 10.1109/robot.2001.932920
- [**O'Rourke 1998**] O'ROURKE, Joseph: *Computational geometry in C*. 2nd. New York, NY, USA : Cambridge University Press, 1998, 161 – 165. – ISBN 0-5216-4010-5
- [**Ott u. a. 2014**] OTT, Sanne ; MOONS, CPH ; KASHIHA, Mohammadamin A. ; BAHR, Claudia ; TUYTTENS, FAM ; BERCKMANS, Daniel ; NIEWOLD, Theo A.: Automated video analysis of pig activity at pen level highly correlates to human observations of behavioural activities. In: *Livestock Science* 160 (2014), 132 – 137
- [**Padia u. Pears 2011**] PADIA, C. ; PEARS, N. E.: A review and characterization of ICP-based symmetry plane localisation in 3D face data / University of York. 2011 (463). – Technical Report
- [**Pan u. a. 2006**] PAN, Gang ; WANG, Yueming ; QI, Yipeng ; WU, Zhaohui: Finding symmetry plane of 3D face shape. In: *Pattern Recognition (ICPR)* Bd. 3, 2006, 1143 – 1146. – DOI 10.1109/ICPR.2006.565. – ISSN 1051-4651
- [**Parihk u. a. 2013**] PARIHK, M. ; PATELY, M. ; BHAT, D.: Animal detection using template matching algorithm. In: *Research in Modern Engineering and Emerging Technology (IJRMEET)* Bd. 1, 2013, 26 – 32.
- [**Parkhi u. a. 2012**] PARKHI, O. M. ; VEDALDI, A. ; ZISSERMAN, A. ; JAWAHAR, C. V.: Cats and dogs. In: *Computer Vision and Pattern Recognition (CVPR)*, 2012
- [**Parkhi u. a. 2011**] PARKHI, O.M. ; VEDALDI, A. ; JAWAHAR, C.V. ; ZISSERMAN, A.: The truth about cats and dogs. In: *Computer Vision (ICCV)*, 2011, 1427 – 1434. – DOI 10.1109/ICCV.2011.6126398. – ISSN 1550-5499
- [**Peng u. a. 2010**] PENG, Yigang ; GANESH, A. ; WRIGHT, J. ; XU, Wenli ; MA, Yi: RASL: Robust alignment by sparse and low-rank decomposition for linearly correlated images. In: *Computer Vision and Pattern Recognition (CVPR)*, 2010, 763 – 770. – DOI 10.1109/CVPR.2010.5540138. – ISSN 1063-6919

- [**Petersen 1922**] PETERSEN, W.E.: The identification of the bovine by means of nose-prints. In: *Dairy Science (JDS)* 5 (1922), Nr. 3, 249 – 258. – DOI 10.3168/jds.S0022-0302(22)94150-5. – ISSN 0022-0302
- [**Phillips u. a. 2005**] PHILLIPS, P. J. ; FLYNN, Patrick J. ; SCRUGGS, Todd ; BOWYER, Kevin W. ; CHANG, Jin ; HOFFMAN, Kevin ; MARQUES, Joe ; MIN, Jaesik ; WOREK, William: Overview of the face recognition grand challenge. In: *Computer Vision and Pattern Recognition (CVPR)* Bd. 1 IEEE, 2005, 947 – 954.
- [**Phillips u. a. 2000**] PHILLIPS, P. J. ; MOON, Hyeonjoon ; RIZVI, Syed A. ; RAUSS, Patrick J.: The FERET evaluation methodology for face-recognition algorithms. In: *Pattern Analysis and Machine Intelligence (TPAMI)* 22 (2000), Nr. 10, 1090 – 1104
- [**Pluk u. a. 2012**] PLUK, A. ; BAHR, C. ; POURSAHERI, A. ; MAERTENS, W. ; NUFFEL, A. van ; BERCKMANS, D.: Automatic measurement of touch and release angles of the fetlock joint for lameness detection in dairy cattle using vision techniques. In: *Dairy Science (JDS)* 95 (2012), Nr. 4, 1738 – 1748. – DOI 10.3168/jds.2011-4547. – ISSN 0022-0302
- [**Poikalainen u. a. 2009**] POIKALAINEN, V ; PRAKS, J ; KOKIN, E ; ALAND, A ; VEERMÄE, I ; PEETS, S ; AHOKAS, J ; PASTELL, M ; HAUTALA, M ; BERCKMANS, D u. a.: Elaboration of basic methods for automatic analysis of cows' gait. In: *International Scientific Conference*, 2009, 216 – 225.
- [**Porto u. a. 2013**] PORTO, Simona M. ; ARCIDIACONO, Claudia ; ANGUZZA, Umberto ; CASCONE, Giovanni: A computer vision-based system for the automatic detection of lying behaviour of dairy cows in free-stall barns. In: *Biosystems Engineering* 115 (2013), Nr. 2, 184 – 194. – DOI 10.1016/j.biosystemseng.2013.03.002. – ISSN 1537-5110
- [**Porto u. a. 2011**] PORTO, Simona M. ; ARCIDIACONO, Claudia ; GUARNERA, Giuseppe C. ; CASCONE, Giovanni: Preliminary study for the implementation of an image analysis algorithm to detect dairy cow presence at the feed barrier. In: *Agricultural Engineering (JAE)* 42 (2011), Nr. 4, 17 – 23. – DOI <http://dx.doi.org/10.4081/jae.2011.4.17>. – ISSN 2239-6268
- [**Porto u. a. 2012**] PORTO, S.M.C. ; ARCIDIACONO, C. ; ANGUZZA, U. ; CASCONE, G.: Detecting cows at the feed barrier by means of an image analysis algorithm. In: *Safety Health and Welfare in Agriculture and in Agro-food Systems (SHWA)* Bd. 5 - Automation, Remote Control, Robot and Innovative Vehicle. Ragusa - Italy : ElleDue Editore, September 2012, 266 – 272 – ISBN 978-8-8905-4734-8
- [**Porto u. a. 2014**] PORTO, S.M.C. ; ARCIDIACONO, C. ; GIUMMARRA, A. ; ANGUZZA, U. ; CASCONE, G.: Localisation and identification performances of a real-time location system based on ultra wide band technology for monitoring

- and tracking dairy cow behaviour in a semi-open free-stall barn. In: *Computers and Electronics in Agriculture (COMPAG)* 108 (2014), Nr. 0, 221 – 229. – DOI 10.1016/j.compag.2014.08.001. – ISSN 0168–1699
- [**Poursaberi u. a. 2010**] POURSABERI, A. ; BAHR, C. ; PLUK, A. ; NUFFEL, A. van ; BERCKMANS, D.: Real-time automatic lameness detection based on back posture extraction in dairy cattle: Shape analysis of cow with image processing techniques. In: *Computers and Electronics in Agriculture (COMPAG)* 74 (2010), Nr. 1, 110 – 119. – DOI 10.1016/j.compag.2010.07.004. – ISSN 0168–1699
- [**Prankl u. a. 2013**] PRANKL, Johann ; ZILLICH, Michael ; VINCZE, Markus: Interactive object modelling based on piecewise planar surface patches. In: *Computer Vision and Image Understanding (CVIU)* 117 (2013), Nr. 6, 718 – 731. – DOI 10.1016/j.cviu.2013.01.010. – ISSN 1077–3142
- [**Prasad 2012**] PRASAD, Dilip K.: Survey of the problem of object detection in real images. In: *Image Processing (IJIP)* 6 (2012), Nr. 6, 441 – 466
- [**Rakesh u. a. 2012**] RAKESH, R. ; SATHYA, R.D. ; VAITHIYANATHN, V.: Human and animal detection systems-[hads] for unauthorized zone. In: *Research Journal of Applied Sciences Engineering and Technology (RJASET)* 4 (2012), Dezember, Nr. 24, 5344 – 5349. – ISSN 2040–7467
- [**Ramanan u. Forsyth 2003**] RAMANAN, Deva ; FORSYTH, David A.: Using temporal coherence to build models of animals. In: *Computer Vision (ICCV)*, 2003, 338 – 345. – DOI 10.1109/ICCV.2003.1238364
- [**Ramanan u. a. 2006**] RAMANAN, Deva ; FORSYTH, David A. ; BARNARD, Kobus: Building models of animals from video. In: *Pattern Analysis and Machine Intelligence (TPAMI)* 28 (2006), Nr. 8, 1319 – 1334. – DOI 10.1109/TPAMI.2006.155
- [**Ramírez-Gutiérrez u. a. 2010**] RAMÍREZ-GUTIÉRREZ, Kelsey ; CRUZ-PÉREZ, Daniel ; PÉREZ-MEANA, Héctor: Face recognition and verification using histogram equalization. In: *Applied Computer Science (ACS)*. Stevens Point, Wisconsin, USA : World Scientific and Engineering Academy and Society (WSEAS), 2010, 85 – 89 – ISBN 978–9–6047–4231–8
- [**Rana u. Flierl 2011**] RANA, Pravin K. ; FLIERL, Markus: View interpolation with structured depth from multiview video. In: *European Signal Processing Conference (EUSIPCO)*. Barcelona, Spain : Ivana Tasic, August 2011, 383 – 387.
- [**Rangdal u. Hanchate 2014**] RANGDAL, Mukesh B. ; HANCHATE, Dinesh B.: Animal detection using histogram oriented gradient. In: *Recent and Innovation Trends in Computing and Communication (IJRITCC)* 2 (2014), Februar, Nr. 2, 178 – 183

- [**Rangelova u. a. 2004**] RANGUELOVA, E. ; HUISKES, M. ; PAUWELS, E.J.: Towards computer-assisted photo-identification of humpback whales. In: *Image Processing (ICIP)* Bd. 3, 2004, 1727 – 1730. – DOI 10.1109/ICIP.2004.1421406. – ISSN 1522–4880
- [**Redd u. a. 2012**] REDD, Thomas C. ; DUBANSKY, Brooke H. ; OSBORN, Michelle L. ; TULLY, Thomas N. ; HOMBERGER, Dominique G.: A registration algorithm for the identification of individual parrots based on the patterns of filing ridges on the internal surface of their upper bill tip. In: *Biometrics and Bioinformatics (IJBB)* 6 (2012), Nr. 3, 68 – 91
- [**Reynolds 2002**] REYNOLDS, Douglas: An overview of automatic speaker recognition. In: *Acoustics, Speech and Signal Processing (ICASSP)*, 2002, 4072 – 4075.
- [**Rojas-Olivares u. a. 2012**] ROJAS-OLIVARES, M. A. ; CAJA, G. ; CARNÉ, S. ; SALAMA, A. A. K. ; ADELL, N. ; PUIG, P.: Determining the optimal age for recording the retinal vascular pattern image of lambs. In: *Animal Science (JAS)* 90 (2012), März, 1040 – 1046. – DOI 10.2527/jas.2010–3648. – ISSN 1525–3163
- [**Rojas-Olivares u. a. 2011**] ROJAS-OLIVARES, M A. ; CAJA, G ; CARNÉ, S ; SALAMA, A A K. ; ADELL, N ; PUIG, P: Retinal image recognition for verifying the identity of fattening and replacement lambs. In: *Animal Science (JAS)* 89 (2011), Februar, Nr. 8, 2603 – 2613. – DOI 10.2527/jas.2010–3197. – ISSN 1525–3163
- [**Rosenfeld u. Pfaltz 1966**] ROSENFELD, Azriel ; PFALTZ, John L.: Sequential operations in digital picture processing. In: *Journal of ACM (JACM)* 13 (1966), Oktober, Nr. 4, 471 – 494. – DOI 10.1145/321356.321357. – ISSN 0004–5411
- [**Rushen u. a. 2012**] RUSHEN, J ; CHAPINAL, N ; PASSILLI, A M de: Automated monitoring of behavioural-based animal welfare indicators. In: *Animal Welfare* 21 (2012), Nr. 3, 339 – 350. – DOI doi:10.7120/09627286.21.3.339
- [**Rusinkiewicz u. Levoy 2001**] RUSINKIEWICZ, Szymon ; LEVOY, Marc: Efficient variants of the ICP algorithm. In: *3D Digital Imaging and Modeling (3DIM)*, 2001. – DOI 10.1109/IM.2001.924423. ISBN 0–7695–0984–3
- [**Sandwell u. a. 2013**] SANDWELL, Roz ; LOOS, Alexander ; BURGHARDT, Tilo: Synthesising unseen image conditions to enhance classification accuracy for sparse datasets: applied to chimpanzee face recognition. In: *British Machine Vision Workshop (BMVW)* (2013). ISBN 1–9017–2550–2
- [**Schatzmann 2012**] SCHATZMANN, U.: Kennzeichnungsmethoden bei Pferden mittels Heissbrand und Transponder-Implantation unter besonderer Berücksichtigung von Schmerzen und Leiden. In: *Ministerium für Landwirtschaft, Umwelt und ländliche Räume* (2012)

- [**Scheenstra u. a. 2005**] SCHEENSTRA, Alize ; RUIFROK, Arnout ; VELTKAMP, Remco C.: A survey of 3D face recognition methods. In: *LNCIS*. Berlin, Heidelberg : Springer, 2005, 891 – 899.
- [**Schiele u. Crowley 2000**] SCHIELE, Bernt ; CROWLEY, JamesL.: Recognition without correspondence using multidimensional receptive field histograms. In: *Computer Vision (IJCV)* 36 (2000), Nr. 1, 31 – 50. – DOI 10.1023/A:1008120406972. – ISSN 0920–5691
- [**Seifzadeh u. Faez 2014**] SEIFZADEH, S. ; FAEZ, K.: A cortex-like model for animal recognition based on texture using feature-selective hashing. In: *Iranian Conference on Intelligent Systems*, 2014. – DOI 10.1109/IranianCIS.2014.6802584
- [**Shanahan u. a. 2010**] SHANAHAN, C. ; AYALEW, G. ; BUTLER, F. ; WARD, S. ; McDONNELL, K.: Implementation protocol utilising radio frequency identification (RFID) and biometric identifiers; In the context of irish bovine traceability. In: TURCU, Cristina (Hrsg.): *Sustainable Radio Frequency Identification Solutions*, INTECH Open Access Publisher, Februar 2010, 105 – 120 – DOI 10.5772/8010. – ISBN 978–9–5376–1974–9
- [**Shao 2003**] SHAO, B.: *A real time imaging system for assessment and control swine thermal comfort*, Iowa State University, PhD thesis, 2003. – Agricultural and Biosystems Engineering
- [**Shao u. Xin 2008**] SHAO, Bin ; XIN, Hongwei: A real-time computer vision assessment and control of thermal comfort for group-housed pigs. In: *Computers and Electronics in Agriculture (COMPAG)* 62 (2008), Nr. 1, 15 – 21. – DOI 10.1016/j.compag.2007.09.006. – ISSN 0168–1699
- [**Shao u. a. 1998**] SHAO, J. ; XIN, H. ; HARMON, J.D.: Comparison of image feature extraction for classification of swine thermal comfort behavior. In: *Computers and Electronics in Agriculture (COMPAG)* 19 (1998), Nr. 3, 223 – 232. – DOI 10.1016/S0168–1699(97)00048–3. – ISSN 0168–1699
- [**Shao u. a. 1997**] SHAO, Junging ; XIN, Hongwei ; HARMON, Jay D.: Neural network analysis of postural behavior of young swine to determine the IR thermal comfort state. In: *Transactions of the ASAE* 40 (1997), Nr. 3, 755 – 760
- [**Sharma u. a. 2014**] SHARMA, S. ; SHAH, D. ; BHAVSAR, R. ; JAISWAL, B.: Automated detection of animals in context to indian scenario. In: *Intelligent Systems, Modelling and Simulation (ISMS)* Bd. 5 UKSim and IEEE, 2014, 334 – 338 – DOI 10.1109/ISMS.2014.63. – ISBN 978–1–4799–3857–5
- [**Sharma 2013**] SHARMA, Sachin: A brief overview on different animal detection methods. In: *Signal & Image Processing (SIPIJ)* 4 (2013), Juni, Nr. 3, 77 – 81. – DOI 10.5121/sipij.2013.4307

- [**Sherley u. a. 2010**] SHERLEY, RB ; BURGHARDT, T ; BARHAM, PJ ; CAMPBELL, N ; CUTHILL, IC: Spotting the difference: towards fully-automated population monitoring of african penguins *spheniscus demersus*. In: *Endangered Species Research (ESR)* 11 (2010), Nr. 2, 101 – 111. – DOI 10.3354/esr00267
- [**Shirley u. Marschner 2009**] SHIRLEY, Peter ; MARSCHNER, Steve: *Fundamentals of computer graphics*. 3rd. Natick, MA, USA : A. K. Peters, Ltd., 2009. – ISBN 978-1-5688-1469-8
- [**Shrivakshan u. Chandrasekar 2011**] SHRIVAKSHAN, G.T. ; CHANDRASEKAR, Dr.C.: Shark fish classification through image processing using wavelet transformation and enhanced edge detection technology. In: *Computer Technology and Applications (IJCTA)* Bd. 2, 2011, 773 – 783. – ISSN 2229-6093
- [**Silberman u. a. 2012**] SILBERMAN, Nathan ; HOIEM, Derek ; KOHLI, Pushmeet ; FERGUS, Rob: Indoor segmentation and support inference from RGBD images. In: *European Conference on Computer Vision (ECCV)*. Berlin, Heidelberg : Springer, 2012, 746 – 760.
- [**Sirovich u. Kirby 1987**] SIROVICH, L. ; KIRBY, M.: Low-dimensional procedure for the characterization of human faces. In: *Optical Society of America A (JOSA A)* 4 (1987), März, Nr. 3, 519 – 524. – DOI 10.1364/JOSAA.4.000519
- [**Smisek u. a. 2011**] SMISEK, J. ; JANCOSSEK, M. ; PAJDLA, T.: 3D with Kinect. In: *Computer Vision (ICCV) Workshops*, 2011, 1154 – 1160. – DOI 10.1109/ICCVW.2011.6130380
- [**Ruiz-del Solar u. Navarrete 2005**] SOLAR, Javier Ruiz-del ; NAVARRETE, Pablo: Eigenspace-based face recognition: A comparative study of different approaches. In: *Systems, Man, and Cybernetics (SMC)* 35 (2005), Nr. 3, 315 – 325
- [**del Solar u. Quinteros 2008**] SOLAR, Javier R. ; QUINTEROS, Julio: Illumination compensation and normalization in eigenspace-based face recognition: A comparative study of different pre-processing approaches. In: *Pattern Recognition Letters (PRL)* 29 (2008), Nr. 14, 1966 – 1979. – DOI 10.1016/j.patrec.2008.06.015. – ISSN 0167-8655
- [**Song u. a. 2008**] SONG, Xiangyu ; LEROY, Toon ; VRANKEN, Erik ; MAERTENS, Willem ; SONCK, Bart ; BERCKMANS, Daniel: Automatic detection of lameness in dairy cattle – Vision-based trackway analysis in cow’s locomotion. In: *Computers and Electronics in Agriculture (COMPAG)* 64 (2008), Nr. 1, 39 – 44. – DOI 10.1016/j.compag.2008.05.016. – ISSN 0168-1699
- [**Soram u. Khomdram 2010**] SORAM, Ranbir ; KHOMDRAM, Memeta: Biometric DNA and ECDLP-based personal authentication system: A superior Posse of security. In: *Computer Science and Network Security (IJCSNS)* 10 (2010), Nr. 1

- [**Spampinato u. a. 2010**] SPAMPINATO, Concetto ; GIORDANO, Daniela ; DI SALVO, Roberto ; CHEN-BURGER, Yun-Heh J. ; FISHER, Robert B. ; NADARAJAN, Gayathri: Automatic fish classification for underwater species behavior understanding. In: *Analysis and Retrieval of Tracked Events and Motion in Imagery Streams (ARTEMIS)*. New York, NY, USA : ACM, 2010, 45 – 50 – DOI 10.1145/1877868.1877881. – ISBN 978–1–4503–0163–3
- [**Spampinato u. a. 2014**] SPAMPINATO, Concetto ; PALAZZO, Simone ; BOOM, Bastian ; OSSENBRUGGEN, Jacco van ; KAVASIDIS, Isaak ; DI SALVO, Roberto ; LIN, Fang-Pang ; GIORDANO, Daniela ; HARDMAN, Lynda ; FISHER, RobertB: Understanding fish behavior during typhoon events in real-life underwater environments. In: *International Journal of Multimedia Tools and Applications* 70 (2014), Nr. 1, 199 – 236. – DOI 10.1007/s11042–012–1101–5. – ISSN 1380–7501
- [**Specht 1988**] SPECHT, Donald F.: Probabilistic neural networks for classification, mapping, or associative memory. In: *Neural Networks (IJCNN)* IEEE, 1988, 525 – 532. – DOI 10.1109/ICNN.1988.23887
- [**Specht 1990**] SPECHT, Donald F.: Probabilistic neural networks. In: *Journal of Neural Networks* 3 (1990), Nr. 1, 109 – 118. – DOI 10.1016/0893–6080(90)90049–Q. – ISSN 0893–6080
- [**Stahl 2007**] STAHL, Henry: *Extraktion des Trifokalen-Tensors aus mehreren Bildern*. Holstenhofweg 85, D-22043 Hamburg, Helmut Schmidt Universität, Universität der Bundeswehr Hamburg, Studienarbeit, Oktober 2007
- [**Stahl u. a. 2012**] STAHL, Henry ; SCHÄDLER, Kristina ; HARTUNG, Eberhard: Erfassung von biometrischen 3D-Daten zur Identifikation von Individuen bei Nutzieren. In: CLASEN, Michael (Hrsg.) ; FRÖHLICH, Georg (Hrsg.) ; BERNHARDT, Heinz (Hrsg.) ; HILDEBRAND, Knut (Hrsg.) ; THEUVSEN, Brigitte (Hrsg.): *32. GIL Jahrestagung* Bd. 194, GI, 2012 (LNI), 287 – 290 – ISBN 978–3–8857–9288–8
- [**Stankovski u. a. 2012**] STANKOVSKI, Stevan ; OSTOJIC, Gordana ; SENK, Ivana ; RAKIC-SKOKOVIC, Marija ; TRIVUNOVIC, Snezana ; KUCEVIC, Denis: Dairy cow monitoring by RFID. In: *Scientia Agricola* 69 (2012), Februar, 75 – 80. – ISSN 0103–9016
- [**Stockman u. Shapiro 2001**] STOCKMAN, George ; SHAPIRO, Linda G.: *Computer vision*. 1st. Upper Saddle River, NJ, USA : Prentice Hall PTR, 2001. – ISBN 0–1303–0796–3
- [**Štruc u. Pavešić 2010**] ŠTRUC, Vitomir ; PAVEŠIĆ, Nikola: The complete gabor-fisher classifier for robust face recognition. In: *Journal on Advances in Signal Processing (EURASIP)* 2010 (2010)

- [**Sugumar u. Jayaparvathy 2014**] SUGUMAR, S. J. ; JAYAPARVATHY, R.: An improved real time image detection system for elephant intrusion along the forest border areas. In: *The Scientific World Journal* 2014 (2014), Januar. – DOI 10.1155/2014/393958
- [**Sumner u. Ross 2012**] SUMNER, Robert ; ROSS, Laura: Extension of the violajones detector – application to cat faces / Boston University. One Silber Way, Boston, MA 02215 : Sumner, Robert and Ross, Laura, Mai 2012 (Computer Vision Class EC500). – Technischer Report
- [**Suzaki u. a. 2001**] SUZAKI, Masahiko ; YAMAKITA, Osamu ; HORIKAWA, Shinichi ; KUNO, Yuji ; AIDA, Hiroko ; SASAKI, Naoki ; KUSUNOSE, Ryo: A horse identification system using biometrics. In: *Systems and Computers in Japan* 32 (2001), Nr. 14, 12 – 23
- [**Taigman u. a. 2014**] TAIGMAN, Yaniv ; YANG, Ming ; RANZATO, Marc'Aurelio ; WOLF, Lior: DeepFace: Closing the gap to human-level performance in face verification. In: *Computer Vision and Pattern Recognition (CVPR)*, 2014
- [**Tang u. a. 2013**] TANG, Shuai ; WANG, Xiaoyu ; LV, Xutao ; HAN, Tony X. ; KELLER, James ; HE, Zhihai ; SKUBIC, Marjorie ; LAO, Shihong: Histogram of oriented normal vectors for object recognition with a depth sensor. In: *Asian Conference on Computer Vision (ACCV)*. Berlin, Heidelberg : Springer, 2013 (Part II), 525 – 538 – DOI 10.1007/978-3-642-37444-9_41. – ISBN 978-3-6423-7443-2
- [**Teillaud 1993**] TEILLAUD, Monique: *LNCS*. Bd. 758: *Towards dynamic randomized algorithms in computational geometry*. Berlin, Heidelberg : Springer, 1993. – ISBN 3-5405-7503-0
- [**Tenenbaum u. a. 2000**] TENENBAUM, Joshua B. ; DE SILVA, Vin ; LANGFORD, John C.: A global geometric framework for nonlinear dimensionality reduction. In: *Science* 290 (2000), Nr. 5500, 2319 – 2323
- [**Thakoor u. Jung 2007**] THAKOOR, Ninad ; JUNG, Sungyong: Real-time planar surface segmentation in disparity space. In: *Computer Vision and Pattern Recognition (CVPR)*, 2007
- [**van Tienhoven u. a. 2007**] TIENHOVEN, J.A. M. ; HARTOG, J. E. ; REIJNS, R. A. ; PEDDEMORS, V. M.: A computer-aided program for pattern-matching of natural marks on the spotted raggedtooth shark *Carcharias taurus*. In: *Applied Ecology* 44 (2007), Nr. 2, 273 – 280. – DOI 10.1111/j.1365-2664.2006.01273.x. – ISSN 1365-2664
- [**Tillett u. a. 2004**] TILLET, RD ; MCFARLANE, NJB ; WU, J ; SCHOFIELD, CP ; JU, X ; SIEBERT, JP: Extracting morphological data from 3D images of pigs. In: *Agricultural Engineering (AgEng)* Bd. 1, 2004, 492 – 493.

- [**Tillett u. a. 1997**] TILLET, R.D. ; ONYANGO, C.M. ; MARCHANT, J.A.: Using model-based image processing to track animal movements. In: *Computers and Electronics in Agriculture (COMPAG)* 17 (1997), Nr. 2, 249 – 261. – DOI 10.1016/S0168-1699(96)01308-7. – ISSN 0168-1699
- [**Timney u. Keil 1999**] TIMNEY, Brian ; KEIL, Kathy: Local and global stereopsis in the horse. In: *Vision Research* 39 (1999), Nr. 10, 1861 – 1867. – DOI 10.1016/S0042-6989(98)00276-4. – ISSN 0042-6989
- [**Tomassini 1996**] TOMASSINI, M.: Evolutionary algorithms. In: SANCHEZ, Eduardo (Hrsg.) ; TOMASSINI, Marco (Hrsg.): *Towards Evolvable Hardware* Bd. 1062. Berlin, Heidelberg : Springer, 1996, 19 – 47. – DOI 10.1007/3-540-61093-6_2. – ISBN 978-3-5406-1093-9
- [**Town u. a. 2013**] TOWN, Christopher ; MARSHALL, Andrea ; SETHASATHIEN, Nutthaporn: Manta matcher: Automated photographic identification of manta rays using keypoint features. In: *Ecology and Evolution (ECE3)* 3 (2013), Nr. 7, 1902 – 1914. – DOI 10.1002/ece3.587. – ISSN 2045-7758
- [**Trucco u. a. 2003**] TRUCCO, E. ; ISGRO, F. ; BRACCHI, F.: Plane detection in disparity space. In: *Visual Information Engineering (VIE)*, 2003, 73 – 76.
- [**Tsai u. Huang 2014**] TSAI, Du-Ming ; HUANG, Ching-Ying: A motion and image analysis method for automatic detection of estrus and mating behavior in cattle. In: *Computers and Electronics in Agriculture (COMPAG)* 104 (2014), Nr. 0, 25 – 31. – DOI 10.1016/j.compag.2014.03.003. – ISSN 0168-1699
- [**Turk u. Pentland 1991a**] TURK, M.A. ; PENTLAND, A.P.: Face recognition using eigenfaces. In: *Computer Vision and Pattern Recognition (CVPR)*, 1991, 586 – 591. – DOI 10.1109/CVPR.1991.139758. – ISSN 1063-6919
- [**Turk u. Pentland 1991b**] TURK, Matthew ; PENTLAND, Alex: Eigenfaces for recognition. In: *Cognitive Neuroscience (JCN)* 3 (1991), Januar, Nr. 1, 71 – 86. – DOI 10.1162/jocn.1991.3.1.71. – ISSN 0898-929X
- [**Tweed u. Calway 2002**] TWEED, D. ; CALWAY, A: Tracking multiple animals in wildlife footage. In: *Pattern Recognition (ICPR)* Bd. 2, 2002, 24 – 27. – DOI 10.1109/ICPR.2002.1048227. – ISSN 1051-4651
- [**Viazzi u. a. 2014**] VIAZZI, S. ; ISMAYILOVA, G. ; OCZAK, M. ; SONODA, L.T. ; FELS, M. ; GUARINO, M. ; VRANKEN, E. ; HARTUNG, J. ; BAHR, C. ; BERCKMANS, D.: Image feature extraction for classification of aggressive interactions among pigs. In: *Computers and Electronics in Agriculture (COMPAG)* 104 (2014), Nr. 0, 57 – 62. – DOI 10.1016/j.compag.2014.03.010. – ISSN 0168-1699
- [**Viola u. Jones 2001a**] VIOLA, P. ; JONES, M.: Rapid object detection using a boosted cascade of simple features. In: *Computer Vision and Pattern Recognition*

- (*CVPR*) Bd. 1, 2001, 511 – 518. – DOI 10.1109/CVPR.2001.990517. – ISSN 1063–6919
- [**Viola u. Jones 2001b**] VIOLA, Paul ; JONES, Michael: Robust real-time object detection. In: *Computer Vision (IJCV)*, 2001, 137 – 154.
- [**Viola u. a. 2003**] VIOLA, Paul ; JONES, Michael ; SNOW, Daniel: Detecting pedestrians using patterns of motion and appearance. In: *Computer Vision (ICCV)* Bd. 2, 2003, 734 – 741 – DOI 10.1109/ICCV.2003.1238422. – ISBN 0–7695–1950–4
- [**Viola u. Jones 2004**] VIOLA, Paul ; JONES, Michael J.: Robust real-time face detection. In: *Computer Vision (IJCV)* 57 (2004), Mai, Nr. 2, 137 – 154. – DOI 10.1023/B:VISI.0000013087.49260.fb. – ISSN 0920–5691
- [**Vosselman u. a. 2004**] VOSSELMAN, G. ; GORTE, B. G. H. ; SITHOLE, G. ; RABBANI, T.: Recognising structure in laser scanner point clouds. In: THIES, M. (Hrsg.) ; KOCH, B. (Hrsg.) ; SPIECKER, H. (Hrsg.) ; WEINACKER, H. (Hrsg.) ; International Society for Photogrammetry and Remote Sensing (Veranst.): *Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS)* Bd. 8/W2 International Society for Photogrammetry and Remote Sensing, 2004, 33 – 38.
- [**Wathes u. a. 2008**] WATHES, CM ; KRISTENSEN, Helle H. ; AERTS, J-M ; BERCKMANS, Daniel: Is precision livestock farming an engineer’s daydream or nightmare, an animal’s friend or foe, and a farmer’s panacea or pitfall? In: *Computers and Electronics in Agriculture (COMPAG)* 64 (2008), Nr. 1, 2 – 10
- [**Wawerla u. a. 2009**] WAWERLA, Jens ; MARSHALL, Shelley ; MORI, Greg ; ROTHLEY, Kristina ; SABZMEYDANI, Payam: BearCam: Automated wildlife monitoring at the arctic circle. In: *Machine Vision and Applications* 20 (2009), Nr. 5, 303 – 317. – DOI 10.1007/s00138–008–0128–0. – ISSN 0932–8092
- [**Weis 2007**] WEIS, Stephen A.: RFID (radio frequency identification): Principles and applications / MIT Computer Science and Artificial Intelligence Lab (CSAIL). 2007. – Forschungsbericht
- [**Werni 1996**] WERNI, Pohl; P.: *Alles für Pferdehalter*. 2. Leopold Stocker Verlag, 1996. – ISBN 978–3–7020–1282–3
- [**Widenhofer 2010**] WIDENHOFER, Robert: Xbox 360 makes a ‘kinectic’ connection. In: *EETimes* Issue 1592 (2010), November, 45 – 46
- [**Wiskott u. a. 1999**] WISKOTT, Laurenz ; FELLOUS, Jean-Marc ; KRÜGER, Norbert ; MALSBURG, Christoph von d.: Face recognition by elastic bunch graph matching. In: JAIN, L. C. (Hrsg.) ; HALICI, U. (Hrsg.) ; HAYASHI, I. (Hrsg.) ; LEE, S. B. (Hrsg.): *Intelligent Biometric Techniques in Fingerprint and Face Recognition*. CRC Press, 1999, 355 – 396. – ISBN 0–8493–2055–0

- [Wong u. a. 2001] WONG, K. W. ; LAM, K. M. ; SIU, W. C.: An efficient algorithm for human face detection and facial feature extraction under different conditions. In: *Pattern Recognition* 34 (2001), Oktober, Nr. 10, 1993 – 2004
- [Wongchoosuk u. a. 2009] WONGCHOOSUK, Chatchawal ; LUTZ, Mario ; KERDCHAROEN, Teerakiat: Detection and classification of human body odor using an electronic nose. In: *Sensors* 9 (2009), Nr. 9, 7234 – 7249
- [Wouters u. a. 1990] WOUTERS, P. ; GEERS, R. ; PARDUYNS, G. ; GOOSSENS, K. ; TRUYEN, B. ; GOEDSEELS, V. ; STUYFT, E. van d.: Image-analysis parameters as inputs for automatic environmental temperature control in piglet houses. In: *Computers and Electronics in Agriculture (COMPAG)* 5 (1990), Nr. 3, 233 – 246. – DOI 10.1016/0168-1699(90)90014-G. – ISSN 0168-1699
- [Wright u. a. 2009] WRIGHT, J. ; YANG, AY. ; GANESH, A ; SASTRY, S.S. ; MA, Yi: Robust face recognition via sparse representation. In: *Pattern Analysis and Machine Intelligence (TPAMI)* 31 (2009), Februar, Nr. 2, 210 – 227. – DOI 10.1109/TPAMI.2008.79. – ISSN 0162-8828
- [Xia u. a. 2011] XIA, Lu ; CHEN, Chia-Chih ; AGGARWAL, J.K.: Human detection using depth information by Kinect. In: *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2011, 15 – 22. – DOI 10.1109/CVPRW.2011.5981811. – ISSN 2160-7508
- [Xin u. Shao 2002] XIN, H. ; SHAO, B.: Real-time assessment of swine thermal comfort by computer vision. In: *World Congress of Computers in Agriculture and Natural Resources ASAE*, 2002 (701P0301), 362 – 369. – DOI 10.13031/2013.8353
- [Xin u. Shao 2005] XIN, H. ; SHAO, B.: Real-time behavior-based assessment and control of swine thermal comfort. In: *International Livestock Environment Symposium (ILES)* 7 (2005), Mai, Nr. 701P0205, 694 – 702. – DOI 10.13031/2013.18431
- [Xin 1999] XIN, Hongwei: Assessing swine thermal comfort by image analysis of postural behaviors. In: *Animal Science (JAS)* 77 (1999), Nr. 2
- [Xu u. a. 2009] XU, Jianyu ; YU, Haizhen ; LIU, Ying: A method to quantify movement activity of groups of animals using automated image analysis. In: *Image Processing and Photonics for Agricultural Engineering (PIAGENG)* 74891C (2009). – DOI 10.1117/12.837187
- [Yamada u. a. 2011] YAMADA, A. ; KOJIMA, K. ; KIYAMA, J. ; OKAMOTO, M. ; MURATA, H.: Directional edge-based dog and cat face detection method for digital camera. In: *Consumer Electronics (ICCE)*, 2011, 87 – 88. – DOI 10.1109/ICCE.2011.5722916. – ISSN 2158-3994

- [**Yang u. a. 2002**] YANG, Ming-Hsuan ; KRIEGMAN, David J. ; AHUJA, Narendra: Detecting faces in images: A survey. In: *Pattern Analysis and Machine Intelligence (TPAMI)* 24 (2002), Januar, Nr. 1, 34 – 58
- [**Yapa u. Harada 2008**] YAPA, Roshan D. ; HARADA, Koichi: Connected component labeling algorithms for gray-scale images and evaluation of performance using digital mammograms. In: *Computer Science and Network Security (IJCSNS)* 8 (2008), Nr. 6, 33 – 41
- [**Yi u. a. 2013**] YI, Dong ; LEI, Zhen ; LI, S.Z.: Towards pose robust face recognition. In: *Computer Vision and Pattern Recognition (CVPR)*, 2013, 3539 – 3545. – DOI 10.1109/CVPR.2013.454. – ISSN 1063-6919
- [**Zeppelzauer 2013**] ZEPPELZAUER, Matthias: Automated detection of elephants in wildlife video. In: *Journal on Image and Video Processing (EURASIP)* 46 (2013), Nr. 1. – DOI 10.1186/1687-5281-2013-46
- [**Zhang u. Zhang 2010**] ZHANG, Cha ; ZHANG, Zhengyou: A survey of recent advances in face detection / Microsoft Research. Redmond, WA 98052, Juni 2010 (MSR-TR-2010-66). – Technical Report
- [**Zhang u. a. 2008**] ZHANG, Weiwei ; SUN, Jian ; TANG, Xiaoou: Cat head detection - how to effectively exploit shape and texture features. In: FORSYTH, David (Hrsg.) ; TORR, Philip (Hrsg.) ; ZISSERMAN, Andrew (Hrsg.): *European Conference on Computer Vision (ECCV)* Bd. 5305. Berlin, Heidelberg : Springer, 2008, 802 – 816. – DOI 10.1007/978-3-540-88693-8_59. – ISBN 978-3-5408-8692-1
- [**Zhang u. a. 2011**] ZHANG, Weiwei ; SUN, Jian ; TANG, Xiaoou: From tiger to panda: Animal head detection. In: *Image Processing (ICIP)* 20 (2011), Juni, Nr. 6, 1696 – 1708. – DOI 10.1109/TIP.2010.2099126. – ISSN 1057-7149
- [**Zhang u. Gao 2009**] ZHANG, Xiaozheng ; GAO, Yongsheng: Face recognition across pose: A review. In: *Pattern Recognition* 42 (2009), Nr. 11, 2876 – 2896. – DOI 10.1016/j.patcog.2009.04.017. – ISSN 0031-3203
- [**Zhang u. a. 2013**] ZHANG, Xin ; YANG, Yee-Hong ; HAN, Zhiguang ; WANG, Hui ; GAO, Chao: Object class detection: A survey. In: *ACM Computing Surveys (CSUR)* 46 (2013), Juli, Nr. 1. – DOI 10.1145/2522968.2522978. – ISSN 0360-0300
- [**Zhang 2004**] ZHANG, Z. ; MEDIONI, G. (Hrsg.) ; KANG, S.B. (Hrsg.): *Camera calibration*. Emerging Topics in Computer Vision. Prentice Hall PTR, 2004, 4 – 43.
- [**Zhang 1999**] ZHANG, Zhengyou: Flexible camera calibration by viewing a plane from unknown orientations. In: *Computer Vision (ICCV)*, 1999, 666 – 673.

- [**Zhao u. a. 2011**] ZHAO, Lindu ; SUN, Shengnan ; WANG, Xiaoqiang: Tracking and traceability system using livestock Iris code in meat supply chain. In: *Innovative Computing, Information and Control (IJICIC)* 7 (2011), Nr. 5A, 2201 – 2212
- [**Zhao u. a. 2003**] ZHAO, W. ; CHELLAPPA, R. ; PHILLIPS, P. J. ; ROSENFELD, A.: Face recognition: A literature survey. In: *ACM Computing Surveys (CSUR)* 35 (2003), Dezember, Nr. 4, 399 – 458. – DOI 10.1145/954339.954342. – ISSN 0360–0300
- [**Zhu 2004**] ZHU, Mu: Recall, precision and average precision / University of Waterloo. ON, N2L 3G1, Canada., August 2004. – Technical Report. – Department of Statistics & Actuarial Science